Sparsity: algorithms, approximations, and analysis

Anna C. Gilbert

Department of Mathematics
University of Michigan
Basic image/signal/data compression: transform coding
Approximate signals sparsely

Compress images, signals, data

accurately (mathematics)
concisely (statistics)
efficiently (algorithms)
If one orthonormal basis is good, surely two (or more) are better...
Redundancy

If one orthonormal basis is good, surely two (or more) are better...

...especially for images
The original image is the sum of the three. Each path and different layers of the original. We can think of the image as being synthesized by these different layers, where each layer is similar to a downsampled version of the original input. This mathematical transform is useful for a more efficient and accurate storage and processing of imaging data, as well as for encoding unidentifiable structure in images. For example, it can sharpen detail in medical images, such as MRI, and be used to identify particular objects for diagnostic purposes.

Sponsored by the
- Joint Policy Board for Mathematics;
- American Mathematical Society
- Mathematical Association of America
- Society for Industrial and Applied Mathematics
- Informax

http://forum.swarthmore.edu/ma wreaks

Images provided by Ronald Coifman, Yale University
**Definition**

A dictionary $D$ in $\mathbb{R}^n$ is a collection $\{\varphi_\ell\}_{\ell=1}^d \subset \mathbb{R}^n$ of unit-norm vectors: $\|\varphi_\ell\|_2 = 1$ for all $\ell$.

- Elements are called *atoms*.
- If $\text{span}\{\varphi_\ell\} = \mathbb{R}^n$, the dictionary is *complete*.
- If $\{\varphi_\ell\}$ are linearly dependent, the dictionary is *redundant*.
Matrix representation

Form a matrix

\[ \Phi = \begin{bmatrix} \varphi_1 & \varphi_2 & \cdots & \varphi_d \end{bmatrix} \]

so that

\[ \Phi c = \sum_{\ell} c_{\ell} \varphi_{\ell}. \]
Examples: Fourier—Dirac

\[ \Phi = [\mathcal{F} | I] \]

\[ \varphi_\ell(t) = \frac{1}{\sqrt{n}} e^{2\pi i \ell t / n} \quad \ell = 1, 2, \ldots, n \]

\[ \varphi_\ell(t) = \delta_\ell(t) \quad \ell = n + 1, n + 2, \ldots, 2n \]
**Sparse Problems**

**Exact.** Given a vector $x \in \mathbb{R}^n$ and a complete dictionary $\Phi$, solve

$$\min_{c} \|c\|_0 \quad \text{s.t.} \quad x = \Phi c$$

i.e., find a sparsest representation of $x$ over $\Phi$.

**Error.** Given $\epsilon \geq 0$, solve

$$\min_{c} \|c\|_0 \quad \text{s.t.} \quad \|x - \Phi c\|_2 \leq \epsilon$$

i.e., find a sparsest approximation of $x$ that achieves error $\epsilon$.

**Sparse.** Given $k \geq 1$, solve

$$\min_{c} \|x - \Phi c\|_2 \quad \text{s.t.} \quad \|c\|_0 \leq k$$

i.e., find the best approximation of $x$ using $k$ atoms.
Theorem
Given an arbitrary redundant dictionary $\Phi$ and a signal $x$, it is NP-hard to solve the sparse representation problem $\text{Sparse}$.
[Natarajan’95,Davis’97]

Corollary

Error is NP-hard as well.

Corollary

It is NP-hard to determine if the optimal error is zero for a given sparsity level $k$. 
**Exact Cover by 3-sets: X3C**

**Definition**
Given a finite universe $\mathcal{U}$, a collection $\mathcal{X}$ of subsets $X_1, X_2, \ldots, X_d$ s.t. $|X_i| = 3$ for each $i$, does $\mathcal{X}$ contain a disjoint collection of subsets whose union $= \mathcal{U}$?

Classic NP-hard problem.

**Proposition**
Any instance of X3C is reducible in polynomial time to SPARSE.

<table>
<thead>
<tr>
<th>$X_1$</th>
<th>$X_2$</th>
<th>$X_3$</th>
<th>$X_N$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\mathcal{U}$</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

...
Bad news, Good news

Bad news

Given any polynomial time algorithm for Sparse, there is a dictionary $\Phi$ and a signal $x$ such that algorithm returns incorrect answer

Pessimistic: worst case

Cannot hope to approximate solution, either
Bad news, Good news

**Bad news**

Given any polynomial time algorithm for \textsc{Sparse}, there is a dictionary \( \Phi \) and a signal \( x \) such that algorithm returns incorrect answer

Pessimistic: worst case

Cannot hope to approximate solution, either

**Good news**

Natural dictionaries are far from arbitrary

Perhaps natural dictionaries admit polynomial time algorithms

Optimistic: rarely see worst case

Leverage our intuition from orthogonal basis
Hardness depends on instance

<table>
<thead>
<tr>
<th>Redundant dictionary $\Phi$</th>
<th>input signal $x$</th>
</tr>
</thead>
<tbody>
<tr>
<td>NP-hard</td>
<td>arbitrary</td>
</tr>
<tr>
<td>depends on choice of $\Phi$</td>
<td>fixed</td>
</tr>
<tr>
<td>compressive sensing</td>
<td>random (distribution?)</td>
</tr>
<tr>
<td>example: spikes and sines</td>
<td>random (distribution?)</td>
</tr>
</tbody>
</table>

random signal model
Sparse algorithms: exploit geometry

Orthogonal case: pull off atoms one at a time, with dot products in decreasing magnitude
Sparse algorithms: exploit geometry

Orthogonal case: pull off atoms one at a time, with dot products in decreasing magnitude
Why is orthogonal case easy?
  inner products between atoms are small
  it’s easy to tell which one is the best choice

When atoms are (nearly) parallel, can’t tell which one is best
Coherence

Definition
The coherence of a dictionary

\[ \mu = \max_{j \neq \ell} | \langle \varphi_j, \varphi_\ell \rangle | \]

Small coherence (good)
Large coherence (bad)
Large, incoherent dictionaries

Fourier–Dirac, $d = 2n, \mu = \frac{1}{\sqrt{n}}$

wavelet packets, $d = n \log n, \mu = \frac{1}{\sqrt{2}}$

There are large dictionaries with coherence close to the lower (Welch) bound; e.g., Kerdock codes, $d = n^2, \mu = 1/\sqrt{n}$
Greedy algorithms

Build approximation one step at a time...

...choose the **best** atom at each step
Orthogonal Matching Pursuit **OMP** [Mallat’93, Davis’97]

**Input.** Dictionary $\Phi$, signal $x$, steps $k$

**Output.** Coefficient vector $c$ with $k$ nonzeros, $\Phi c \approx x$

**Initialize.** counter $t = 1$, $c = 0$

1. **Greedy selection.**

   \[ l_t = \arg \max |\Phi^*(x - \Phi c)| \]

2. **Update.** Find $c_{l_1}, \ldots, c_{l_t}$ to solve

   \[
   \min \left\| x - \sum_{s} c_{l_s} \varphi_{l_s} \right\|_2
   \]

   new approximation $a_t \leftarrow \Phi c$

3. **Iterate.** $t \leftarrow t + 1$, stop when $t > k$. 
Many greedy algorithms with similar outline

Matching Pursuit: replace step 2. by
\[ c_{\ell t} \leftarrow c_{\ell t} + \langle x - \Phi c, \varphi_{\ell t} \rangle \]

Thresholding
Choose \( m \) atoms where \( |\langle x, \varphi_{\ell} \rangle| \) are among \( m \) largest

Alternate stopping rules:
\[ \|x - \Phi c\|_2 \leq \epsilon \]
\[ \max_{\ell} |\langle x - \Phi c, \varphi_{\ell} \rangle| \leq \epsilon \]

Many other variations
Convergence of OMP

Theorem

Suppose $\Phi$ is a complete dictionary for $\mathbb{R}^n$. For any vector $x$, the residual after $t$ steps of OMP satisfies

$$\|x - \Phi c\|_2 \leq \frac{C}{\sqrt{t}}.$$ 

[Devore-Temlyakov]
Convergence of OMP

Theorem

Suppose $\Phi$ is a complete dictionary for $\mathbb{R}^n$. For any vector $x$, the residual after $t$ steps of OMP satisfies

$$\|x - \Phi c\|_2 \leq \frac{C}{\sqrt{t}}.$$

[Devore-Temlyakov]

Even if $x$ can be expressed sparsely, OMP may take $n$ steps before the residual is zero.

But, sometimes OMP correctly identifies sparse representations.
Theorem (ERC)
A sufficient condition for OMP to identify $\Lambda$ after $k$ steps is that

$$\max_{\ell \notin \Lambda} \| \Phi^+_\Lambda \varphi_\ell \|_1 < 1$$

where $A^+ = (A^* A)^{-1} A^*$.

Theorem
The ERC holds whenever $k < \frac{1}{2} (\mu^{-1} + 1)$. Therefore, OMP can recover any sufficiently sparse signals.

For most redundant dictionaries, $k < \frac{1}{2} (\sqrt{n} + 1)$. 

[Tropp'04]
Sparse representation with OMP

Suppose $x$ has $k$-sparse representation

\[
x = \sum_{\ell \in \Lambda} b_{\ell} \varphi_{\ell}\quad \text{where } |\Lambda| = k
\]

Sufficient to find $\Lambda$—When can OMP do so?

Define

\[
\Phi_{\Lambda} = \begin{bmatrix}
\varphi_{\ell_1} & \varphi_{\ell_2} & \cdots & \varphi_{\ell_k}
\end{bmatrix}_{\ell_s \in \Lambda}
\]

and

\[
\Psi_{\Lambda} = \begin{bmatrix}
\varphi_{\ell_1} & \varphi_{\ell_2} & \cdots & \varphi_{\ell_{N-k}}
\end{bmatrix}_{\ell_s \notin \Lambda}
\]

Define \textit{greedy selection ratio}

\[
\rho(r) = \frac{\max_{\ell \notin \Lambda} |\langle r, \varphi_{\ell} \rangle|}{\max_{\ell \in \Lambda} |\langle r, \varphi_{\ell} \rangle|} = \frac{\|\Psi_{\Lambda}^* r\|_{\infty}}{\|\Phi_{\Lambda}^* r\|_{\infty}} = \frac{\max \text{ i.p. bad atoms}}{\max \text{ i.p. good atoms}}
\]

OMP chooses good atom iff $\rho(r) < 1$
Theorem
Assume $k \leq \frac{1}{3\mu}$. For any vector $x$, the approximation $\hat{x}$ after $k$ steps of OMP satisfies

$$\|x - \hat{x}\|_2 \leq \sqrt{1 + 6k} \|x - x_k\|_2$$

where $x_k$ is the best $k$-term approximation to $x$. [Tropp’04]

Theorem
Assume $4 \leq k \leq \frac{1}{\sqrt{\mu}}$. Two-phase greedy pursuit produces $\hat{x}$ s.t.

$$\|x - \hat{x}\|_2 \leq 3 \|x - x_k\|_2.$$  

Assume $k \leq \frac{1}{\mu}$. Two-phase greedy pursuit produces $\hat{x}$ s.t.

$$\|x - \hat{x}\|_2 \leq \left(1 + \frac{2\mu k^2}{\left(1 - 2\mu k\right)^2}\right) \|x - x_k\|_2.$$

[Gilbert, Strauss, Muthukrishnan, Tropp’03]
Alternative algorithmic approach

**Exact:** non-convex optimization

$$\min ||c||_0 \quad \text{s.t.} \quad x = \Phi c$$
Alternative algorithmic approach

**Exact**: non-convex optimization

\[
\min \|c\|_0 \quad \text{s.t.} \quad x = \Phi c
\]

Convex relaxation of non-convex problem

\[
\min \|c\|_1 \quad \text{s.t.} \quad x = \Phi c
\]

**Error**: non-convex optimization

\[
\arg \min \|c\|_0 \quad \text{s.t.} \quad \|x - \Phi c\|_2 \leq \epsilon
\]
Alternative algorithmic approach

**Exact**: non-convex optimization

\[
\min \|c\|_0 \quad \text{s.t.} \quad x = \Phi c
\]

Convex relaxation of non-convex problem

\[
\min \|c\|_1 \quad \text{s.t.} \quad x = \Phi c
\]

**Error**: non-convex optimization

\[
\arg \min \|c\|_0 \quad \text{s.t.} \quad \|x - \Phi c\|_2 \leq \epsilon
\]

Convex relaxation of non-convex problem

\[
\arg \min \|c\|_1 \quad \text{s.t.} \quad \|x - \Phi c\|_2 \leq \delta.
\]
Convex relaxation: algorithmic formulation

Well-studied algorithmic formulation [Donoho, Donoho-Elad-Temlyakov, Tropp, and many others]

Optimization problem = linear program: linear objective function (with variables $c^+$, $c^-$) and linear or quadratic constraints

Still need algorithm for solving optimization problem

Hard part of analysis: showing solution to convex problem = solution to original problem
Exact Recovery Condition

**Theorem (ERC)**

A sufficient condition for BP to recover the sparsest representation of $x$ is that

$$\max_{\ell \notin \Lambda} \| \Phi_\Lambda^+ \varphi_\ell \|_1 < 1$$

where $A^+ = (A^T A)^{-1} A^T$. [Tropp’04]
Theorem (ERC)

A sufficient condition for BP to recover the sparsest representation of $x$ is that

$$\max_{\ell \notin \Lambda} \| \Phi^+_{\Lambda} \varphi_\ell \|_1 < 1$$

where $A^+ = (A^T A)^{-1} A^T$. [Tropp'04]

Theorem

The ERC holds whenever $k < \frac{1}{2}(\mu^{-1} + 1)$. Therefore, BP can recover any sufficiently sparse signals. [Tropp'04]
Alternate optimization formulations

Constrained minimization:

\[
\arg \min_{c} \|c\|_1 \quad \text{s.t.} \quad \|x - \Phi c\|_2 \leq \delta.
\]

Unconstrained minimization:

\[
\minimize L(c; \gamma, x) = \frac{1}{2} \|x - \Phi c\|_2^2 + \gamma \|c\|_1.
\]

Many algorithms for \(\ell_1\)-regularization
Sparse approximation: Optimization vs. Greedy

**Exact** and **Error** amenable to convex relaxation and convex optimization

**Sparse** not amenable to convex relaxation

\[
\arg \min \| \Phi c - x \|_2 \quad \text{s.t.} \quad \|c\|_0 \leq k
\]

*but* appropriate for greedy algorithms
Connection between...

Sparse Approximation and Statistical Learning
Sparsity in statistical learning

\[ X_j = (X_{1j}, \ldots, X_{N_j})^T \]

**Goal:** Given \( X \) and \( y \), find \( \alpha \) and coeffs. \( \beta \in \mathbb{R}^p \) for linear model that minimizes the error

\[
(\hat{\alpha}, \hat{\beta}) = \arg \min \|X\beta - (y - \alpha)\|_2^2.
\]

**Solution:**

Least squares: low bias but large variance and hard to interpret
lots of non-zero coefficients
Shrink \( \beta_j \)'s, make \( \beta \) sparse.
Algorithms in statistical learning

**Brute force:** Calculate Mallows’ $C_p$ for every subset of predictor variables, and choose the best one.

**Greedy algorithms:** Forward selection, forward stagewise, least angle regression (LARS), backward elimination.

**Constrained optimization:** Quadratic programming problem with linear constraints (e.g., LASSO).

**Unconstrained optimization:** regularization techniques

Sparse approximation and SVM equivalence [Girosi '96]
Connection between...

Sparse Approximation and Compressed Sensing
Interchange roles

\[ \Phi \text{ data/image measurements/coeffs.} \]
Problem statement

Construct

Matrix $\Phi: \mathbb{R}^n \rightarrow \mathbb{R}^m$

Decoding algorithm $\mathcal{D}$

Given $\Phi x$ for any signal $x \in \mathbb{R}^n$, we can, with high probability, quickly recover $\hat{x}$ with

$$\|x - \hat{x}\|_p \leq (1 + \epsilon) \min_{y \ k-sparse} \|x - y\|_q = (1 + \epsilon)\|x - x_k\|_q$$

Assume $x$ has low complexity: $x$ is $k$-sparse (with noise)
Comparison with Sparse Approximation

**Sparse:** Given $y$ and $\Phi$, find (sparse) $x$ such that $y = \Phi x$. Return $\hat{x}$ with guarantee

$$\|\Phi \hat{x} - y\|_2 \quad \text{small compared with} \quad \|y - \Phi x_k\|_2.$$ 

**CS:** Given $y$ and $\Phi$, find (sparse) $x$ such that $y = \Phi x$. Return $\hat{x}$ with guarantee

$$\|\hat{x} - x\|_p \quad \text{small compared with} \quad \|x - x_k\|_q.$$ 

$p$ and $q$ not always the same, not always $= 2$. 
**Analogy: root-finding**

\[ \hat{p} \text{ with } |f(\hat{p}) - 0| \leq \epsilon \]

\[ \hat{p} \text{ with } |\hat{p} - p| \leq \epsilon \]

**Sparse:** Given \( f \) (and \( y = 0 \)), find \( p \) such that \( f(p) = 0 \). Return \( \hat{p} \) with guarantee

\[ |f(\hat{p}) - 0| \text{ small.} \]

**CS:** Given \( f \) (and \( y = 0 \)), find \( p \) such that \( f(p) = 0 \). Return \( \hat{p} \) with guarantee

\[ |\hat{p} - p| \text{ small.} \]
Parameters

Number of measurements $m$
Recovery time
Approximation guarantee (norms, mixed)
One matrix vs. distribution over matrices
Explicit construction
Universal matrix (for any basis, after measuring)
Tolerance to measurement noise
Applications

Data stream algorithms
\[ x_i = \text{number of items with index } i \]
can maintain \( \Phi x \) under increments to \( x \)
recover approximation to \( x \)

Efficient data sensing
digital/analog cameras
analog-to-digital converters
high throughput biological screening
(pooling designs)

Error-correcting codes
code \( \{ y \in \mathbb{R}^n \mid \Phi y = 0 \} \)
\( x = \text{error vector, } \Phi x = \text{syndrome} \)
Two approaches

**Geometric** [Donoho '04], [Candes-Tao '04, '06], [Candes-Romberg-Tao '05], [Rudelson-Vershynin '06], [Cohen-Dahmen-DeVore '06], and many others...

*Dense* recovery matrices that satisfy RIP (e.g., Gaussian, Fourier)

Geometric recovery methods ($\ell_1$ minimization, LP)

$$\hat{x} = \text{argmin}\|z\|_1 \text{ s.t. } \Phi z = \Phi x$$

Uniform guarantee: one matrix $A$ that works for all $x$

**Combinatorial** [Gilbert-Guha-Indyk-Kotidis-Muthukrishnan-Strauss '02], [Charikar-Chen-FarachColton '02], [Cormode-Muthukrishnan '04], [Gilbert-Strauss-Tropp-Vershynin '06, '07]

*Sparse* random matrices (typically)

Combinatorial recovery methods or weak, greedy algorithms

Per-instance guarantees, later uniform guarantees
Summary

Sparse approximation, statistical learning, and compressive sensing intimately related

Many models of computation and scientific/technological problems in which they all arise

Algorithms for all similar: optimization and greedy

Community progress on geometric and statistical models for matrices $\Phi$ and signals $x$, different problem instance types

Explicit constructions?

Better/different geometric/statistical models?

Better connections with coding and complexity theory?