Algorithms for sparse analysis

Lecture II: Hardness results for sparse approximation problems

Anna C. Gilbert

Department of Mathematics
University of Michigan
Complexity theory: Reductions

- Problem $A$ (efficiently) reduces to $B$ means a(n efficient) solution to $B$ can be used to solve $A$ (efficiently).
- If we have an algorithm to solve $B$, then we can use that algorithm to solve $A$; i.e., $A$ is easier to solve than $B$.
- “reduces” does not confer simplification here.

- Definition

  $A \leq_P B$ if there’s polynomial time computable function $f$ s.t.

  $$w \in A \iff f(w) \in B.$$ 

  - $B$ at least as hard as $A$.
Complexity theory: **NP-hard**

- **Definition**
  \( A \in \text{NP-complete} \) if (i) \( A \in \text{NP} \) and (ii) for all \( X \in \text{NP} \), \( X \leq_P A \).

- **Definition**
  \( B \in \text{NP-hard} \) if there is \( A \in \text{NP-complete} \) s.t. \( A \leq_P B \).
Examples

- **RelPrime** Are $a$ and $b$ relatively prime?
  - in $P$
    - Euclidean algorithm, simple

- **Primes** Is $x$ a prime number?
  - in $P$
    - highly non-trivial algorithm, does not determine factors

- **Factor** Factor $x$ as a product of powers of primes.
  - in $NP$
    - not known to be **NP-hard**

- **X3C** Given a finite universe $\mathcal{U}$, a collection $\mathcal{X}$ of subsets $X_1, X_2, \ldots, X_N$ s.t. $|X_i| = 3$ for each $i$, does $\mathcal{X}$ contain a disjoint collection of subsets whose union $= \mathcal{U}$?
  - **NP-complete**
Theorem
Given an arbitrary redundant dictionary \( \Phi \), a signal \( x \), and a sparsity parameter \( k \), it is NP-hard to solve the sparse representation problem \( \text{D-Exact} \). [Natarajan’95, Davis’97]
NP-hardness

**Theorem**

*Given an arbitrary redundant dictionary $\Phi$, a signal $x$, and a sparsity parameter $k$, it is NP-hard to solve the sparse representation problem $D$-EXACT.*  [Natarajan’95, Davis’97]

**Corollary**

$\text{Sparse, Error, Exact}$ are all NP-hard.
Theorem

Given an arbitrary redundant dictionary $\Phi$, a signal $x$, and a sparsity parameter $k$, it is NP-hard to solve the sparse representation problem $D$-$\text{EXACT}$. [Natarajan’95, Davis’97]

Corollary

Sparse, Error, Exact are all NP-hard.

Corollary

Given an arbitrary redundant dictionary $\Phi$ and a signal $x$, it is NP-hard to approximate (in error) the solution of $\text{EXACT}$ to within any factor. [Davis’97]
Exact Cover by 3-sets: $X_3C$

Definition
Given a finite universe $\mathcal{U}$, a collection $\mathcal{X}$ of subsets $X_1, X_2, \ldots, X_N$ s.t. $|X_i| = 3$ for each $i$, does $\mathcal{X}$ contain a disjoint collection of subsets whose union $= \mathcal{U}$?
Exact Cover by 3-sets: \textit{X3C}

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

\textbf{NP-complete} problem.
Exact Cover by 3-sets: $X3C$

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

$\textbf{NP-complete}$ problem.

Proposition

Any instance of X3C is reducible in polynomial time to D-EXACT. X3C \leq_P D-EXACT

Proof.

- Let \( \Omega = \{1, 2, \ldots, N\} \) index \( \Phi \). Set \( \varphi_i = 1_{X_i} \).

Select \( x = (1, 1, \ldots, 1) \), \( k = \frac{1}{3} |\mathcal{U}| \).

- Suppose have solution to X3C. Sufficient to check if Sparse solution has zero error.

Assume solutions of X3C indexed by \( \Lambda \). Set \( c_{opt} = 1_{\Lambda} \).

\( \Phi c_{opt} = x \).

\( \Rightarrow \) Sparse solution has zero error and D-Exact returns Yes.
Proposition

Any instance of $X3C$ is reducible in polynomial time to $D$-Exact. $X3C \leq_p D$-Exact

Proof.

- Let $\Omega = \{1, 2, \ldots, N\}$ index $\Phi$. Set $\varphi_i = 1_{X_i}$.
  Select $x = (1, 1, \ldots, 1)$, $k = \frac{1}{3} |\mathcal{U}|$.

- Suppose have solution to $X3C$. Sufficient to check if Sparse solution has zero error.
  Assume solutions of $X3C$ indexed by $\Lambda$. Set $c_{opt} = 1_\Lambda$.
  \[ \Phi c_{opt} = x. \]
  \[ \implies \text{Sparse solution has zero error and D-Exact returns YES}. \]

- Suppose $c_{opt}$ is optimal solution of Sparse
  \[ \Phi c_{opt} = x \]
  then $c_{opt}$ contains $k \leq \frac{1}{3} |\mathcal{U}|$ nonzero entries and D-Exact returns YES.
  Each column of $\Phi$ has 3 nonzero entries
  \[ \implies \{X_i \mid i \in \text{supp}(c_{opt})\} \text{ is disjoint collection covering } \mathcal{U}. \]
What does this mean?

Bad news

• Given any polynomial time algorithm for $\text{Sparse}$, there is a dictionary $\Phi$ and a signal $x$ such that algorithm returns incorrect answer
• Pessimistic: worst case
• Cannot hope to approximate solution, either
What does this mean?

Bad news

- Given any polynomial time algorithm for $\text{SPARSE}$, 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
- Hardness depends on instance type
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>
</tbody>
</table>

random signal model
Leverage intuition from orthonormal basis

- Suppose $\Phi$ is orthogonal, $\Phi^{-1} = \Phi^T$
- Solution to EXACT problem is unique

$$c = \Phi^{-1}x = \Phi^T x \quad \text{i.e.,} \quad c_\ell = \langle x, \varphi_\ell \rangle$$

hence, $x = \sum_\ell \langle x, \varphi_\ell \rangle \varphi_\ell$. 
Leverage intuition from orthonormal basis

Solution to Sparse problem similar

- Let $\ell_1 \leftarrow \text{arg max}_\ell |\langle x, \varphi_\ell \rangle|$
  
  Set $c_{\ell_1} \leftarrow \langle x, \varphi_{\ell_1} \rangle$
  
  Residual $r \leftarrow x - c_{\ell_1} \varphi_{\ell_1}$
Leverage intuition from orthonormal basis

Solution to Sparse problem similar

- Let $\ell_1 \leftarrow \arg \max_{\ell} |\langle x, \varphi_{\ell} \rangle|$
  
  Set $c_{\ell_1} \leftarrow \langle x, \varphi_{\ell_1} \rangle$
  
  Residual $r \leftarrow x - c_{\ell_1} \varphi_{\ell_1}$

- Let $\ell_2 \leftarrow \arg \max_{\ell} |\langle r, \varphi_{\ell} \rangle| = \arg \max_{\ell} |\langle x - c_{\ell_1} \varphi_{\ell_1}, \varphi_{\ell} \rangle| = \arg \max_{\ell \neq \ell_1} |\langle x, \varphi_{\ell} \rangle|$
  
  Set $c_{\ell_2} \leftarrow \langle r, \varphi_{\ell_2} \rangle$.
  
  Update residual $r \leftarrow x - (c_{\ell_1} \varphi_{\ell_1} + c_{\ell_2} \varphi_{\ell_2})$
Leverage intuition from orthonormal basis

Solution to \textsc{Sparse} problem similar

\begin{itemize}
\item Let $\ell_1 \leftarrow \arg\max_{\ell} |\langle x, \varphi_\ell \rangle|$

Set $c_{\ell_1} \leftarrow \langle x, \varphi_{\ell_1} \rangle$

Residual $r \leftarrow x - c_{\ell_1} \varphi_{\ell_1}$

\item Let $\ell_2 \leftarrow \arg\max_{\ell} |\langle r, \varphi_\ell \rangle| = \arg\max_{\ell} |\langle x - c_{\ell_1} \varphi_{\ell_1}, \varphi_\ell \rangle| = \arg\max_{\ell \neq \ell_1} |\langle x, \varphi_\ell \rangle|$

Set $c_{\ell_2} \leftarrow \langle r, \varphi_{\ell_2} \rangle$.

Update residual $r \leftarrow x - (c_{\ell_1} \varphi_{\ell_1} + c_{\ell_2} \varphi_{\ell_2})$

\item Repeat $k - 2$ times.
\end{itemize}
Leverage intuition from orthonormal basis

Solution to $\text{Sparse}$ problem similar

- Let $\ell_1 \leftarrow \arg \max_{\ell} |\langle x, \varphi_\ell \rangle|$
  
  Set $c_{\ell_1} \leftarrow \langle x, \varphi_{\ell_1} \rangle$

  Residual $r \leftarrow x - c_{\ell_1}\varphi_{\ell_1}$

- Let $\ell_2 \leftarrow \arg \max_{\ell} |\langle r, \varphi_\ell \rangle| = \arg \max_{\ell} |\langle x - c_{\ell_1}\varphi_{\ell_1}, \varphi_\ell \rangle| = \arg \max_{\ell \neq \ell_1} |\langle x, \varphi_\ell \rangle|$
  
  Set $c_{\ell_2} \leftarrow \langle r, \varphi_{\ell_2} \rangle$.

  Update residual $r \leftarrow x - (c_{\ell_1}\varphi_{\ell_1} + c_{\ell_2}\varphi_{\ell_2})$

- Repeat $k - 2$ times.

- Set $c_{\ell} \leftarrow 0$ for $\ell \neq \ell_1, \ell_2, \ldots, \ell_k$. 

Approximate $x \approx \Phi c = \sum_{t=1}^{k} \langle x, \varphi_{\ell_t} \rangle \varphi_{\ell_t}$.

Check: algorithm generates list of coeffs of $x$ over basis in descending order (by absolute value).
Leverage intuition from orthonormal basis

Solution to Sparse problem similar

- Let $\ell_1 \leftarrow \text{arg max}_\ell |\langle x, \varphi_\ell \rangle|$
  Set $c_{\ell_1} \leftarrow \langle x, \varphi_{\ell_1} \rangle$
  Residual $r \leftarrow x - c_{\ell_1} \varphi_{\ell_1}$

- Let $\ell_2 \leftarrow \text{arg max}_\ell |\langle r, \varphi_\ell \rangle| = \text{arg max}_\ell |\langle x - c_{\ell_1} \varphi_{\ell_1}, \varphi_\ell \rangle| = \text{arg max}_{\ell \neq \ell_1} |\langle x, \varphi_\ell \rangle|$
  Set $c_{\ell_2} \leftarrow \langle r, \varphi_{\ell_2} \rangle$.
  Update residual $r \leftarrow x - (c_{\ell_1} \varphi_{\ell_1} + c_{\ell_2} \varphi_{\ell_2})$

- Repeat $k - 2$ times.
- Set $c_\ell \leftarrow 0$ for $\ell \neq \ell_1, \ell_2, \ldots, \ell_k$.
- Approximate $x \approx \Phi c = \sum_{t=1}^{k} \langle x, \varphi_{\ell_t} \rangle \varphi_{\ell_t}$.

Check: algorithm generates list of coeffs of $x$ over basis in descending order (by absolute value).
Geometry

• Why is orthogonal case easy?
  inner products between atoms are small
  it’s easy to tell which one is the best choice

\[ \langle r, \varphi_j \rangle = \langle x - c_i \varphi_i, \varphi_j \rangle = \langle x, \varphi_j \rangle - c_i \langle \varphi_i, \varphi_j \rangle \]

• 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)
Coherence

Definition

The coherence of a dictionary

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

Small coherence (good)  Large coherence (bad)
Coherence: lower bound

Theorem

For a $d \times N$ dictionary,

$$\mu \geq \sqrt{\frac{N - d}{d(N - 1)}} \approx \frac{1}{\sqrt{d}}.$$

[Welch'73]

Theorem

For most pairs of orthonormal bases in $\mathbb{R}^d$, the coherence between the two is

$$\mu = O\left(\sqrt{\frac{\log d}{d}}\right).$$

[Donoho, Huo '99]
Large, incoherent dictionaries

- Fourier–Dirac, $N = 2d$, $\mu = \frac{1}{\sqrt{d}}$
- wavelet packets, $N = d \log d$, $\mu = \frac{1}{\sqrt{2}}$
- There are large dictionaries with coherence close to the lower (Welch) bound; e.g., Kerdock codes, $N = d^2$, $\mu = 1/\sqrt{d}$
Approximation algorithms (error)

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

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

i.e., find the best approximation of $x$ using $k$ atoms.

- $c_{opt} = \text{optimal solution}$
- $E_{opt} = \|\Phi c_{opt} - x\|_2 = \text{optimal error}$
Approximation algorithms (error)

- **Sparse.** Given $k \geq 1$, solve
  \[
  \arg \min_c \| x - \Phi c \|_2 \quad \text{s.t.} \quad \| c \|_0 \leq k
  \]
  i.e., find the best approximation of $x$ using $k$ atoms.

- $c_{opt} = \text{optimal solution}$
- $E_{opt} = \| \Phi c_{opt} - x \|_2 = \text{optimal error}$

- Algorithm returns $\hat{c}$ with
  
  1. $\| \hat{c} \|_0 = k$
  2. $E = \| \Phi \hat{c} - x \|_2 \leq C_1 E_{opt}$
Approximation algorithms (error)

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

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

  i.e., find the best approximation of $x$ using $k$ atoms.

- $c_{opt} = \text{optimal solution}$
- $E_{opt} = \|\Phi c_{opt} - x\|_2 = \text{optimal error}$

- Algorithm returns $\hat{c}$ with

  1. $\|\hat{c}\|_0 = k$
  2. $E = \|\Phi \hat{c} - x\|_2 \leq C_1 E_{opt}$

- (Error) approximation ratio: $\frac{E}{E_{opt}} = \frac{C_1 E_{opt}}{E_{opt}} = C_1$
Approximation algorithms (terms)

- Algorithm returns $\hat{c}$ with
  
  \begin{align*}
  (1) \quad & \|\hat{c}\|_0 = C_2 k \\
  (2) \quad & E = \|\Phi\hat{c} - x\|_2 = E_{opt}
  \end{align*}

- (Terms) approximation ratio: $\frac{\|\hat{c}\|_0}{\|c_{opt}\|_0} = \frac{C_2 k}{k} = C_2$
Algorithm returns $\hat{c}$ with

\begin{align*}
(1) \quad & \|\hat{c}\|_0 = C_2 k \\
(2) \quad & E = \|\Phi\hat{c} - x\|_2 = C_1 E_{opt}
\end{align*}

(Terms, Error) approximation ratio: $(C_2, C_1)$
Greedy algorithms

Build approximation one step at a time...
Greedy algorithms

Build approximation one step at a time...

...choose the best atom at each step
Orthogonal Matching Pursuit OMP [Mallat '92], [Davis'97]

**Input.** Dictionary Φ, signal x, steps k

**Output.** Coefficient vector c with k nonzeros, Φc ≈ x
Orthogonal Matching Pursuit OMP [Mallat '92], [Davis'97]

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

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

**Initialize.** counter $t = 1$, residual $r_0 = x$, $c = 0$
Orthogonal Matching Pursuit \textbf{OMP} [Mallat '92], [Davis'97]

\textbf{Input.} Dictionary $\Phi$, signal $x$, steps $k$
\textbf{Output.} Coefficient vector $c$ with $k$ nonzeros, $\Phi c \approx x$
\textbf{Initialize.} counter $t = 1$, residual $r_0 = x$, $c = 0$

1. \textbf{Greedy selection.} Find atom $\varphi_{j_t}$ s.t.

\[ j_t = \arg\max_\ell | \langle r_{t-1}, \varphi_\ell \rangle | \]
Orthogonal Matching Pursuit (OMP) [Mallat '92], [Davis'97]

**Input.** Dictionary Φ, signal x, steps k

**Output.** Coefficient vector c with k nonzeros, Φc ≈ x

**Initialize.** counter \( t = 1 \), residual \( r_0 = x \), \( c = 0 \)

1. **Greedy selection.** Find atom \( \varphi_{j_t} \) s.t.
   
   \[ j_t = \arg\max_{\ell} |\langle r_{t-1}, \varphi_{\ell} \rangle| \]

2. **Update.** Find \( c_{\ell_1}, \ldots, c_{\ell_t} \) to solve
   
   \[
   \min \left\| x - \sum_{s} c_{\ell_s} \varphi_{\ell_s} \right\|_2
   \]
Orthogonal Matching Pursuit OMP [Mallat '92], [Davis'97]

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

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

**Initialize.** counter $t = 1$, residual $r_0 = x$, $c = 0$

1. **Greedy selection.** Find atom $\varphi_{j_t}$ s.t.

$$ j_t = \arg \max_{\ell} | \langle r_{t-1}, \varphi\ell \rangle | $$

2. **Update.** Find $c_{\ell_1}, \ldots, c_{\ell_t}$ to solve

$$ \min \left\| x - \sum_s c_s \varphi_s \right\|_2 $$

new residual $r_t \leftarrow x - \Phi c$
Orthogonal Matching Pursuit OMP [Mallat '92], [Davis'97]

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

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

**Initialize.** counter $t = 1$, residual $r_0 = x$, $c = 0$

1. **Greedy selection.** Find atom $\varphi_{j_t}$ s.t.

   $$j_t = \arg\max_{\ell} |\langle r_{t-1}, \varphi_\ell \rangle|$$

2. **Update.** Find $c_{\ell_1}, \ldots, c_{\ell_t}$ to solve

   $$\min \left\| x - \sum_s c_s \varphi_s \right\|_2$$

   new residual $r_t \leftarrow x - \Phi c$

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

- **Matching Pursuit**: replace step 2. by $c_{t+1} \leftarrow c_t + \langle r_{t-1}, \varphi_{k_t} \rangle$

- **Thresholding**
  Choose $m$ atoms where $|\langle x, \varphi_\ell \rangle|$ are among $m$ largest

- **Alternate stopping rules**:
  $$\|r_t\|_2 \leq \epsilon$$
  $$\max_\ell |\langle r_t, \varphi_\ell \rangle| \leq \epsilon$$

- **Many other variations**
Summary

- Sparse approximation problems are **NP-hard**
- At least as hard as other well-studied problems
- Hardness result of arbitrary input: *dictionary and signal*
- Intuition from orthonormal basis suggests some feasible solutions under certain conditions on redundant dictionary
- Geometric properties and greedy algorithms
- **Next lecture**: rigorous proofs for algorithms