The Effectiveness of Convex Programming in the Information and Physical Sciences

Emmanuel Candès

Simons Institute Open Lecture, UC Berkeley, October 2013
Today I want to tell you three stories from my life. That's it. No big deal. Just three stories.

Steve Jobs often have missing information:

1. Missing phase (phase retrieval)
2. Missing and/or corrupted entries in data matrix (robust PCA)
3. Missing high-frequency spectrum (super-resolution)

Makes signal/data recovery difficult.

This lecture:

Convex programming usually (but not always) returns the right answer!
Today I want to tell you three stories from my life. That’s it. No big deal. Just three stories.
Three stories

Today I want to tell you three stories from my life. That’s it. No big deal. Just three stories

Steve Jobs

Often have missing information:

(1) Missing phase (phase retrieval)
(2) Missing and/or corrupted entries in data matrix (robust PCA)
(3) Missing high-frequency spectrum (super-resolution)

Makes signal/data recovery difficult

This lecture

Convex programming usually (but not always) returns the right answer!
Story # 1: Phase Retrieval

Collaborators: Y. Eldar, X. Li, T. Strohmer, V. Voroninski
X-ray crystallography

Method for determining atomic structure within a crystal

principle

10 Nobel Prizes in X-ray crystallography, and counting...
Importance

principle

Franklin’s photograph
Missing phase problem

Detectors only record intensities of diffracted rays

→ magnitude measurements only!

Fraunhofer diffraction → intensity of electrical field

\[
|\hat{x}(f_1, f_2)|^2 = \left| \int x(t_1, t_2) e^{-i2\pi(f_1 t_1 + f_2 t_2)} \, dt_1 dt_2 \right|^2
\]
Missing phase problem

Detectors only record intensities of diffracted rays

→ magnitude measurements only!

Fraunhofer diffraction → intensity of electrical field

\[
|\hat{x}(f_1, f_2)|^2 = \left| \int x(t_1, t_2) e^{-i2\pi(f_1 t_1 + f_2 t_2)} \, dt_1 \, dt_2 \right|^2
\]

Phase retrieval problem (inversion)

How can we recover the phase (or equivalently signal \(x(t_1, t_2)\)) from \(|\hat{x}(f_1, f_2)|\)?
About the importance of phase...
About the importance of phase...
About the importance of phase...
About the importance of phase...
X-ray imaging: now and then

Röntgen (1895)

Dierolf (2010)
Ultrashort X-ray pulses

Imaging single large protein complexes
Discrete mathematical model

- Phaseless measurements about $x_0 \in \mathbb{C}^n$

$$ b_k = |\langle a_k, x_0 \rangle|^2 \quad k \in \{1, \ldots, m\} = [m] $$

- Phase retrieval is feasibility problem

$$ \text{find} \quad x $$
$$ \text{subject to} \quad |\langle a_k, x \rangle|^2 = b_k \quad k \in [m] $$

- Solving quadratic equations is NP hard in general
Discrete mathematical model

- Phaseless measurements about $x_0 \in \mathbb{C}^n$

\[ b_k = |\langle a_k, x_0 \rangle|^2 \quad k \in \{1, \ldots, m\} = [m] \]

- Phase retrieval is feasibility problem

\[
\text{find } x \\
\text{subject to } |\langle a_k, x \rangle|^2 = b_k \quad k \in [m]
\]

- Solving quadratic equations is NP hard in general

Nobel Prize for Hauptman and Karle ('85): make use of very specific prior knowledge
Discrete mathematical model

- Phaseless measurements about $x_0 \in \mathbb{C}^n$
  
  $$b_k = |\langle a_k, x_0 \rangle|^2 \quad k \in \{1, \ldots, m\} = [m]$$

- Phase retrieval is feasibility problem

  $$\begin{array}{l}
  \text{find } x \\
  \text{subject to } |\langle a_k, x \rangle|^2 = b_k \quad k \in [m]
  \end{array}$$

- Solving quadratic equations is NP hard in general

Nobel Prize for Hauptman and Karle ('85): make use of very specific prior knowledge

Standard approach: Gerchberg Saxton (or Fienup) iterative algorithm

- Sometimes works well
- Sometimes does not
Quadratic equations: geometric view I
Quadratic equations: geometric view I
Quadratic equations: geometric view I
Quadratic equations: geometric view I
Quadratic equations: geometric view 1
Quadratic equations: geometric view 1
Quadratic equations: geometric view II
PhaseLift

\[ |\langle a_k, x \rangle|^2 = b_k \quad k \in [m] \]
PhaseLift

\[ |\langle a_k, x \rangle|^2 = b_k \quad k \in [m] \]

\textbf{Lifting: } \quad X = xx^*

\[ |\langle a_k, x \rangle|^2 = \text{Tr}(x^* a_k a_k^* x) = \text{Tr}(a_k a_k^* xx^*) := \text{Tr}(A_k X) \quad a_k a_k^* = A_k \]

Turns quadratic measurements into linear measurements about \( xx^* \)
PhaseLift

\[ |\langle a_k, x \rangle|^2 = b_k \quad k \in [m] \]

**Lifting:** \( X = xx^* \)

\[ |\langle a_k, x \rangle|^2 = \text{Tr}(x^* a_k a_k^* x) = \text{Tr}(a_k a_k^* xx^*) := \text{Tr}(A_k X) \quad a_k a_k^* = A_k \]

Turns quadratic measurements into linear measurements about \( xx^* \)

---

**Phase retrieval: equivalent formulation**

| find \( X \) | \( \text{Tr}(A_k X) = b_k \quad k \in [m] \) | \( \min \text{ rank}(X) \) |
| s. t. \( X \succeq 0, \text{rank}(X) = 1 \) | \( \iff \) s. t. \( \text{Tr}(A_k X) = b_k \quad k \in [m] \) |
| \( X \succeq 0 \) |

Combinatorially hard
PhaseLift

\[ |\langle a_k, x \rangle|^2 = b_k \quad k \in [m] \]

**Lifting:** \( X = xx^* \)

\[ |\langle a_k, x \rangle|^2 = \text{Tr}(x^* a_k a_k^* x) = \text{Tr}(a_k a_k^* xx^*) := \text{Tr}(A_k X) \quad a_k a_k^* = A_k \]

Turns quadratic measurements into linear measurements about \( xx^* \)

---

**PhaseLift: tractable semidefinite relaxation**

\[
\begin{align*}
\text{minimize} & \quad \text{Tr}(X) \\
\text{subject to} & \quad \text{Tr}(A_k X) = b_k \quad k \in [m] \\
& \quad X \succeq 0
\end{align*}
\]

- This is a semidefinite program (SDP)
- Trace is a convex proxy for rank
Semidefinite programming (SDP)

- Special class of convex optimization problems
- Relatively natural extension of linear programming (LP)
- ‘Efficient’ numerical solvers (interior point methods)

### LP (std. form): \( x \in \mathbb{R}^n \)

- minimize \( \langle c, x \rangle \)
- subject to \( a_k^T x = b_k \; \; k = 1, \ldots \)
- \( x \geq 0 \)

### SDP (std. form): \( X \in \mathbb{R}^{n \times n} \)

- minimize \( \langle C, X \rangle \)
- subject to \( \langle A_k, X \rangle = b_k \; \; k = 1, \ldots \)
- \( X \succeq 0 \)

Standard inner product: \( \langle C, X \rangle = \text{Tr}(C^* X) \)
From overdetermined to highly underdetermined

Quadratic equations

\[ b_k = |\langle a_k, x \rangle|^2 \]
\[ k \in [m] \]

Lift

\[ b = A(xx^*) \]

minimize

subject to

\[ \text{Tr}(X) = b \]
\[ X \succeq 0 \]

Have we made things worse?

overdetermined \((m > n)\) \quad \rightarrow \quad \text{highly underdetermined} \quad (m \ll n^2)
This is not really new...

Relaxation of quadratically constrained QP’s
  • Shor (87) [Lower bounds on nonconvex quadratic optimization problems]
  • Goemans and Williamson (95) [MAX-CUT]
  • Ben-Tal and Nemirovskii (01) [Monograph]
  • ...

Similar approach for array imaging: Chai, Moscoso, Papanicolaou (11)
Exact phase retrieval via SDP

Quadratic equations

\[ b_k = |\langle a_k, x \rangle|^2 \quad k \in [m] \quad b = A(xx^*) \]

Simplest model: \( a_k \) independently and uniformly sampled on unit sphere

- of \( \mathbb{C}^n \) if \( x \in \mathbb{C}^n \) (complex-valued problem)
- of \( \mathbb{R}^n \) if \( x \in \mathbb{R}^n \) (real-valued problem)
Exact phase retrieval via SDP

Quadratic equations

\[ b_k = |\langle a_k, x \rangle|^2 \quad k \in [m] \quad b = A(xx^*) \]

Simplest model: \( a_k \) independently and uniformly sampled on unit sphere

- of \( \mathbb{C}^n \) if \( x \in \mathbb{C}^n \) (complex-valued problem)
- of \( \mathbb{R}^n \) if \( x \in \mathbb{R}^n \) (real-valued problem)

Theorem (C. and Li ('12); C., Strohmer and Voroninski ('11))

Assume \( m \gtrsim n \). With prob. \( 1 - O(e^{-\gamma m}) \), for all \( x \in \mathbb{C}^n \), only point in feasible set

\[ \{ X : A(X) = b \text{ and } X \succeq 0 \} \text{ is } xx^* \]
Exact phase retrieval via SDP

Quadratic equations

\[ b_k = |\langle a_k, x \rangle|^2 \quad k \in [m] \quad b = A(xx^*) \]

Simplest model: \( a_k \) independently and uniformly sampled on unit sphere

- of \( \mathbb{C}^n \) if \( x \in \mathbb{C}^n \) (complex-valued problem)
- of \( \mathbb{R}^n \) if \( x \in \mathbb{R}^n \) (real-valued problem)

**Theorem (C. and Li ('12); C., Strohmer and Voroninski ('11))**

Assume \( m \gtrsim n \). With prob. \( 1 - O(e^{-\gamma m}) \), for all \( x \in \mathbb{C}^n \), only point in feasible set

\[ \{ X : A(X) = b \quad \text{and} \quad X \succeq 0 \} \]

is \( xx^* \)

Injectivity if \( m \geq 4n - 2 \) (Balan, Bodmann, Casazza, Edidin '09)
How is this possible?

How can feasible set \( \{ X \succeq 0 \} \cap \{ \mathcal{A}(X) = b \} \) have a unique point?

Intersection of \( \begin{bmatrix} x & y \\ y & z \end{bmatrix} \succeq 0 \) with affine space
Correct representation

Rank-1 matrices are on the boundary (extreme rays) of PSD cone
My mental representation

\[ xx^* + H \not\subseteq 0 \]
My mental representation

$xx^* + H \not\subseteq 0$
My mental representation

\[ xx^* + H \not\subseteq 0 \]
My mental representation
My mental representation
Extensions to physical setups

Random masking + diffraction

Similar theory: C., Li and Soltanolkotabi ('13)
Numerical results: noiseless recovery

(a) Smooth signal (real part)  
(b) Random signal (real part)

Figure: Recovery (with reweighting) of $n$-dimensional complex signal ($2n$ unknowns) from $4n$ quadratic measurements (random binary masks)
With noise

\[ b_k \approx |\langle x, a_k \rangle|^2 \quad k \in [m] \]

Noise aware recovery (SDP)

minimize \[ \|A(X) - b\|_1 = \sum_k |\text{Tr}(a_k a_k^* X) - b_k| \]
subject to \[ X \succeq 0 \]

Signal \( \hat{x} \) obtained by extracting first eigenvector (PC) of solution matrix
With noise

\[ b_k \approx \left| \langle x, a_k \rangle \right|^2 \quad k \in [m] \]

**Noise aware recovery (SDP)**

\[
\begin{align*}
\text{minimize} & & \| A(X) - b \|_1 = \sum_k | \text{Tr}(a_k a_k^* X) - b_k | \\
\text{subject to} & & X \succeq 0
\end{align*}
\]

Signal \( \hat{x} \) obtained by extracting first eigenvector (PC) of solution matrix

In same setup as before and for realistic noise models, no method whatsoever can possibly yield a fundamentally smaller recovery error [C. and Li (2012)]
Numerical results: noisy recovery

Figure: SNR versus relative MSE on a dB-scale for different numbers of illuminations with binary masks
Numerical results: noiseless 2D images

Original image

3 Gaussian masks

8 binary masks

Error with 8 binary masks

Courtesy S. Marchesini (LBL)
Story #2: Robust Principal Component Analysis

Collaborators: X. Li, Y. Ma, J. Wright
The separation problem (Chandrasekaran et al.)

\[ M = L + S \]

- \( M \): data matrix (observed)
- \( L \): low-rank (unobserved)
- \( S \): sparse (unobserved)
The separation problem (Chandrasekaran et al.)

\[ M = L + S \]

- \( M \): data matrix (observed)
- \( L \): low-rank (unobserved)
- \( S \): sparse (unobserved)

Problem: can we recover \( L \) and \( S \) accurately?

Again, missing information
Motivation: robust principal component analysis (RPCA)

PCA sensitive to outliers: breaks down with one (badly) corrupted data point
Motivation: robust principal component analysis (RPCA)

PCA sensitive to outliers: breaks down with one (badly) corrupted data point

\[
\begin{bmatrix}
  x_{11} & x_{12} & \cdots & x_{1n} \\
  x_{21} & x_{22} & \cdots & x_{2n} \\
  \vdots & \vdots & \ddots & \vdots \\
  x_{d1} & x_{d2} & \cdots & x_{dn}
\end{bmatrix}
\]

\[\Rightarrow\]

\[
\begin{bmatrix}
  x_{11} & x_{12} & \cdots & x_{1n} \\
  x_{21} & x_{22} & \cdots & x_{2n} \\
  \vdots & \vdots & \ddots & \vdots \\
  x_{d1} & x_{d2} & \cdots & x_{dn}
\end{bmatrix}
\]
Robust PCA

- Data increasingly high dimensional

- Gross errors frequently occur in many applications
  - Image processing
  - Web data analysis
  - Bioinformatics
  - ...  
  - Occlusions
  - Malicious tampering
  - Sensor failures
  - ...

Important to make PCA robust
Gross errors

Observe corrupted entries

\[ Y_{ij} = L_{ij} + S_{ij} \quad (i, j) \in \Omega_{\text{obs}} \]

- \( L \) low-rank matrix
- \( S \) entries that have been tampered with (impulsive noise)

Problem

Recover \( L \) from missing and corrupted samples
The L+S model

(Partial) information \( y = A(M) \) about

\[
\begin{align*}
M &= L + S \\
\text{object} &\quad \text{low rank} & \quad \text{sparse}
\end{align*}
\]

- RPCA
  
  \[ \begin{bmatrix}
  \times & ? & ? & ? & \times & ? \\
  \times & ? & ? & \times & ? & ? \\
\end{bmatrix} \]

  \text{data} = \text{low-dimensional structure} + \text{corruption}

- Dynamic MR

  \text{video seq.} = \text{static background} + \text{sparse innovation}

- Graphical modeling with hidden variables: Chandrasekaran, Sanghavi, Parrilo, Willsky ('09, '11)

  \text{marginal inverse covariance of observed variables} = \text{low-rank} + \text{sparse}
When does separation make sense?

\[ M = L + S \]

Low-rank component cannot be sparse: 

\[ L = \begin{bmatrix}
  * & * & * & * & \cdots & * & * \\
  0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
  \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
  0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
\end{bmatrix} \]
When does separation make sense?

\[ M = L + S \]

Low-rank component cannot be sparse: \( L = \begin{bmatrix}
* & * & * & * & \cdots & * & * \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 
\end{bmatrix} \)

Sparse component cannot be low rank: \( S = \begin{bmatrix}
* & 0 & 0 & 0 & \cdots & 0 & 0 \\
* & 0 & 0 & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
* & 0 & 0 & 0 & \cdots & 0 & 0 
\end{bmatrix} \)
Low-rank component cannot be sparse

\[ L = \begin{bmatrix}
* & * & * & * & \cdots & * & * \\
* & * & * & * & \cdots & * & * \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
\end{bmatrix} \]

Incoherent condition [C. and Recht ('08)]: column and row spaces not aligned with coordinate axes (singular vectors are not sparse)
Low-rank component cannot be sparse

\[
M = \begin{bmatrix}
* & * & \text{skull} & * & \cdots & * & * \\
* & * & \text{skull} & * & \cdots & * & \text{skull} \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0
\end{bmatrix}
\]

Incoherent condition [C. and Recht ('08)]: column and row spaces not aligned with coordinate axes (singular vectors are not sparse)
Sparse component cannot be low-rank

\[
L = \begin{bmatrix}
  x_1 & x_2 & \cdots & x_{n-1} & x_n \\
  x_1 & x_2 & \cdots & x_{n-1} & x_n \\
  \vdots & \vdots & \ddots & \vdots & \vdots \\
  x_1 & x_2 & \cdots & x_{n-1} & x_n \\
\end{bmatrix}
\]

\[\Rightarrow L + S = \begin{bmatrix}
  x_2 & \cdots & x_{n-1} & x_n \\
  x_2 & \cdots & x_{n-1} & x_n \\
  \vdots & \vdots & \ddots & \vdots \\
  x_2 & \cdots & x_{n-1} & x_n \\
\end{bmatrix}
\]

Sparsity pattern will be assumed (uniform) random
Demixing by convex programming

\[ M = L + S \]

- \( L \) unknown (rank unknown)
- \( S \) unknown (\# of entries \( \neq 0 \), locations, magnitudes all unknown)
Demixing by convex programming

\[ M = L + S \]

- \( L \) unknown (rank unknown)
- \( S \) unknown (\# of entries \( \neq 0 \), locations, magnitudes all unknown)

Recovery via SDP

\[
\begin{align*}
\text{minimize} & \quad \|\hat{L}\|_* + \lambda \|\hat{S}\|_1 \\
\text{subject to} & \quad \hat{L} + \hat{S} = M
\end{align*}
\]

See also Chandrasekaran, Sanghavi, Parrilo, Willsky ('09)

- nuclear norm: \( \|L\|_* = \sum_i \sigma_i(L) \) (sum of sing. values)
- \( \ell_1 \) norm: \( \|S\|_1 = \sum_{ij} |S_{ij}| \) (sum of abs. values)
Exact recovery via SDP

$$\min \|\hat{L}\|_* + \lambda \|\hat{S}\|_1 \quad \text{s. t.} \quad \hat{L} + \hat{S} = M$$
Exact recovery via SDP

\[
\min \| \hat{L} \|_* + \lambda \| \hat{S} \|_1 \quad \text{s. t.} \quad \hat{L} + \hat{S} = M
\]

**Theorem**

- \( L \) is \( n \times n \) of \( \text{rank}(L) \leq \rho_r n (\log n)^{-2} \) and incoherent
- \( S \) is \( n \times n \), random sparsity pattern of cardinality at most \( \rho_s n^2 \)

Then with probability \( 1 - O(n^{-10}) \), SDP with \( \lambda = 1/\sqrt{n} \) is exact:

\[ \hat{L} = L, \quad \hat{S} = S \]

Same conclusion for rectangular matrices with \( \lambda = 1/\sqrt{\text{max dim}} \).
Exact recovery via SDP

\[
\min \|\hat{L}\|_* + \lambda \|\hat{S}\|_1 \quad \text{s. t.} \quad \hat{L} + \hat{S} = M
\]

**Theorem**

- \(L\) is \(n \times n\) of \(\text{rank}(L) \leq \rho_r n (\log n)^{-2}\) and incoherent
- \(S\) is \(n \times n\), random sparsity pattern of cardinality at most \(\rho_s n^2\)

Then with probability \(1 - O(n^{-10})\), SDP with \(\lambda = 1/\sqrt{n}\) is exact:

\[
\hat{L} = L, \quad \hat{S} = S
\]

Same conclusion for rectangular matrices with \(\lambda = 1/\sqrt{\max \text{dim}}\)

- No tuning parameter!
- Whatever the magnitudes of \(L\) and \(S\)
Phase transitions in probability of success

(a) PCP, Random Signs

(b) PCP, Coherent Signs

(c) Matrix Completion

$L = X Y^T$ is a product of independent $n \times r$ i.i.d. $\mathcal{N}(0, 1/n)$ matrices
Missing *and* corrupted

**RPCA**

\[
\begin{align*}
\min & \quad \|\hat{L}\|_* + \lambda\|\hat{S}\|_1 \\
\text{s. t.} & \quad \hat{L}_{ij} + \hat{S}_{ij} = L_{ij} + S_{ij} \quad (i, j) \in \Omega_{\text{obs}}
\end{align*}
\]
RPCA

$$\begin{align*}
\min & \quad \|\hat{L}\|_* + \lambda\|\hat{S}\|_1 \\
\text{s. t.} & \quad \hat{L}_{ij} + \hat{S}_{ij} = L_{ij} + S_{ij} \quad (i, j) \in \Omega_{\text{obs}}
\end{align*}$$

Theorem

- $L$ as before
- $\Omega_{\text{obs}}$ random set of size $0.1n^2$ (missing frac. is arbitrary)
- Each observed entry corrupted with prob. $\tau \leq \tau_0$

Then with prob. $1 - O(n^{-10})$, PCP with $\lambda = 1/\sqrt{0.1n}$ is exact:

$$\hat{L} = L$$

Same conclusion for rectangular matrices with $\lambda = 1/\sqrt{0.1\text{max dim}}$
Background subtraction
With noise

With Li, Ma, Wright & Zhou ('10)

Z stochastic or deterministic perturbation

\[ Y_{ij} = L_{ij} + S_{ij} + Z_{ij} \quad (i, j) \in \Omega \]

\[
\begin{align*}
\text{minimize} & \quad ||\hat{L}||_* + \lambda ||\hat{S}||_1 \\
\text{subject to} & \quad \sum_{(i,j) \in \Omega} (M_{ij} - \hat{L}_{ij} - \hat{S}_{ij})^2 \leq \delta^2
\end{align*}
\]

When perfect (noiseless) separation occurs \(\Rightarrow\) noisy variant is stable
Story #3: Super-resolution

Collaborator: C. Fernandez-Granda
Limits of resolution

In any optical imaging system, *diffraction* imposes fundamental limit on resolution.

\[ \delta(t - \tau) \quad \text{optical system} \quad h(t - \tau) \]

*The physical phenomenon called diffraction is of the utmost importance in the theory of optical imaging systems* (Joseph Goodman)
Bandlimited imaging systems (Fourier optics)

\[ f_{\text{obs}}(t) = (h \ast f)(t) \]

\[ \hat{f}_{\text{obs}}(\omega) = \hat{h}(\omega) \hat{f}(\omega) \]

\( h \) : point spread function (PSF)

\( \hat{h} \) : transfer function (TF)

### Bandlimited system

\[ |\omega| > \Omega \quad \Rightarrow \quad |\hat{h}(\omega)| = 0 \]

\[ \hat{f}_{\text{obs}}(\omega) = \hat{h}(\omega) \hat{f}(\omega) \rightarrow \text{suppresses all high-frequency components} \]
Bandlimited imaging systems (Fourier optics)

\[
f_{\text{obs}}(t) = (h \ast f)(t) \quad \quad \quad h : \text{ point spread function (PSF)}
\]

\[
\hat{f}_{\text{obs}}(\omega) = \hat{h}(\omega) \hat{f}(\omega) \quad \quad \quad \hat{h} : \text{ transfer function (TF)}
\]

Bandlimited system

\[
|\omega| > \Omega \Rightarrow |\hat{h}(\omega)| = 0
\]

\[
\hat{f}_{\text{obs}}(\omega) = \hat{h}(\omega) \hat{f}(\omega) \rightarrow \text{suppresses all high-frequency components}
\]

Example: coherent imaging

\[
\hat{h}(\omega) = 1_P(\omega) \quad \text{indicator of pupil element}
\]

TF
Pupil

PSF
Airy disk

cross-section (PSF)
Rayleigh resolution limit

Lord Rayleigh
The super-resolution problem

Retrieve fine scale information from low-pass data

Equivalent description: extrapolate spectrum (ill posed)
Random vs. low-frequency sampling

Random sampling (CS)  
Low-frequency sampling (SR)

Compressive sensing: spectrum \textit{interpolation}  
Super-resolution: spectrum \textit{extrapolation}
Super-resolving point sources

Signal of interest is superposition of point sources
- Celestial bodies in astronomy
- Line spectra in speech analysis
- Fluorescent molecules in single-molecule microscopy

Many applications
- Radar
- Spectroscopy
- Medical imaging
- Astronomy
- Geophysics
- ...
Single molecule imaging (with WE Moerner’s Lab)

Microscope receives light from fluorescent molecules

Problem
Resolution is much coarser than size of individual molecules (low-pass data)
Can we ‘beat’ the diffraction limit and super-resolve those molecules?

Higher molecule density $\rightarrow$ faster imaging
Mathematical model

- Signal

\[ x = \sum_j a_j \delta_{\tau_j} \]
\[ a_j \in \mathbb{C}, \tau_j \in T \subset [0, 1] \]

- Data \( y = \mathcal{F}_n x \): \( n = 2f_{lo} + 1 \) low-frequency coefficients (Nyquist sampling)

\[ y(k) = \int_0^1 e^{-i2\pi kt} x(dt) = \sum_j a_j e^{-i2\pi k\tau_j} \quad k \in \mathbb{Z}, |k| \leq f_{lo} \]

- Resolution limit: \( (\lambda_{lo}/2 \text{ is Rayleigh distance}) \)

\[ 1/f_{lo} = \lambda_{lo} \]
Mathematical model

- Signal

\[ x = \sum_j a_j \delta_{\tau_j} \]
\[ a_j \in \mathbb{C}, \tau_j \in T \subset [0, 1] \]

- Data \( y = \mathcal{F}_n x: n = 2f_{\text{lo}} + 1 \) low-frequency coefficients (Nyquist sampling)

\[ y(k) = \int_0^1 e^{-i2\pi kt} x(dt) = \sum_j a_j e^{-i2\pi k\tau_j} \quad k \in \mathbb{Z}, |k| \leq f_{\text{lo}} \]

- Resolution limit: \((\lambda_{\text{lo}}/2 \text{ is Rayleigh distance})\)

\[ 1/f_{\text{lo}} = \lambda_{\text{lo}} \]

Question

Can we resolve the signal beyond this limit?

Swap time and frequency \(\longrightarrow\) spectral estimation
Can you find the spikes?

Low-frequency data about spike train
Can you find the spikes?

Low-frequency data about spike train
Recovery by minimum total-variation

Recovery by cvx prog.

\[
\begin{align*}
\min & \quad \| \tilde{x} \|_{TV} \\
\text{subject to} & \quad F_n \tilde{x} = y
\end{align*}
\]

\[
\| x \|_{TV} = \int |x(dt)| \quad \text{is continuous analog of } \ell_1 \text{ norm}
\]

\[
x = \sum_j a_j \delta_{\tau_j} \implies \| x \|_{TV} = \sum_j |a_j|
\]

With noise

\[
\min \frac{1}{2} \| y - F_n \tilde{x} \|_{\ell_2}^2 + \lambda \| \tilde{x} \|_{TV}
\]
Recovery by convex programming

\[ y(k) = \int_{0}^{1} e^{-i2\pi kt} x(t) \, dt \quad |k| \leq f_{lo} \]
Recovery by convex programming

\[ y(k) = \int_0^1 e^{-i2\pi kt} x(dt) \quad |k| \leq f_{lo} \]

Theorem (C. and Fernandez Granda (2012))

If spikes are separated by at least

\[ 2 / f_{lo} := 2 \lambda_{lo} \]

then min TV solution is exact! For real-valued \( x \), a min dist. of \( 1.87 \lambda_{lo} \) suffices
Recovery by convex programming

\[ y(k) = \int_0^1 e^{-i2\pi kt} x(dt) \quad |k| \leq f_{lo} \]

**Theorem (C. and Fernandez Granda (2012))**

*If spikes are separated by at least*

\[ 2 / f_{lo} := 2 \lambda_{lo} \]

*then min TV solution is exact! For real-valued x, a min dist. of \(1.87\lambda_{lo}\) suffices*

- Infinite precision!
Recovery by convex programming

\[ y(k) = \int_0^1 e^{-i2\pi kt} x(dt) \quad |k| \leq f_{lo} \]

Theorem (C. and Fernandez Granda (2012))

*If spikes are separated by at least*

\[ 2 / f_{lo} := 2 \lambda_{lo} \]

*then min TV solution is exact!* For real-valued \( x \), a min dist. of \( 1.87 \lambda_{lo} \) suffices

- Infinite precision!
- Whatever the amplitudes
Recovery by convex programming

\[ y(k) = \int_0^1 e^{-i2\pi kt} x(dt) \quad |k| \leq f_{lo} \]

**Theorem (C. and Fernandez Granda (2012))**

*If spikes are separated by at least*

\[ 2 / f_{lo} := 2 \lambda_{lo} \]

*then min TV solution is exact! For real-valued x, a min dist. of 1.87\(\lambda_{lo}\) suffices*

- Infinite precision!
- Whatever the amplitudes
- Can recover \((2\lambda_{lo})^{-1} = f_{lo}/2 = n/4\) spikes from \(n\) low-freq. samples
Recovery by convex programming

\[ y(k) = \int_0^1 e^{-i2\pi kt} x(dt) \quad |k| \leq f_{lo} \]

**Theorem (C. and Fernandez Granda (2012))**

*If spikes are separated by at least*

\[ 2 / f_{lo} := 2\lambda_{lo} \]

*then min TV solution is exact! For real-valued \( x \), a min dist. of \( 1.87\lambda_{lo} \) suffices*

- Infinite precision!
- Whatever the amplitudes
- Can recover \( (2\lambda_{lo})^{-1} = f_{lo}/2 = n/4 \) spikes from \( n \) low-freq. samples
- Cannot go below \( \lambda_{lo} \)
Recovery by convex programming

\[ y(k) = \int_0^1 e^{-i2\pi kt} x(dt) \quad |k| \leq f_{lo} \]

**Theorem (C. and Fernandez Granda (2012))**

*If spikes are separated by at least*

\[ 2/f_{lo} := 2\lambda_{lo} \]

*then min TV solution is exact! For real-valued \( x \), a min dist. of \( 1.87\lambda_{lo} \) suffices*

- Infinite precision!
- Whatever the amplitudes
- Can recover \( (2\lambda_{lo})^{-1} = f_{lo}/2 = n/4 \) spikes from \( n \) low-freq. samples
- Cannot go below \( \lambda_{lo} \)
- Essentially same result in higher dimensions
About separation: sparsity is not enough!

- CS: sparse signals are ‘away’ from null space of sampling operator
- Super-res: this is not the case
About separation: sparsity is not enough!

- CS: sparse signals are ‘away’ from null space of sampling operator
- Super-res: this is not the case
Analysis via prolate spheroidal functions

If distance between spikes less than $\lambda_{lo}/2$ (Rayleigh), problem hopelessly ill posed.
# Formulation as a finite-dimensional problem

### Primal problem

$$\min \, \|x\|_{TV} \quad \text{s.t.} \quad F_n x = y$$

- Infinite-dimensional variable $x$
- Finitely many constraints

### Dual problem

$$\max \, \text{Re}\langle y, c \rangle \quad \text{s.t.} \quad \|F_n^* c\|_{\infty} \leq 1$$

- Finite-dimensional variable $c$
- Infinitely many constraints

\[(F_n^* c)(t) = \sum_{|k| \leq f_{\text{lo}}} c_k e^{i2\pi kt}\]
Formulation as a finite-dimensional problem

<table>
<thead>
<tr>
<th>Primal problem</th>
<th>Dual problem</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \min |x|_{TV} \text{ s. t. } \mathcal{F}_n x = y )</td>
<td>( \max \operatorname{Re}\langle y, c \rangle \text{ s. t. } |\mathcal{F}<em>n^* c|</em>\infty \leq 1 )</td>
</tr>
</tbody>
</table>

- Infinite-dimensional variable \( x \)
- Finitely many constraints

- Finite-dimensional variable \( c \)
- Infinitely many constraints

\[
(\mathcal{F}_n^* c)(t) = \sum_{|k| \leq f_{lo}} c_k e^{i2\pi kt}
\]

Semidefinite representability

\(|(\mathcal{F}_n^* c)(t)| \leq 1 \text{ for all } t \in [0, 1] \text{ equivalent to}\

1. there is \( Q \) Hermitian s. t.
   \[
   \begin{bmatrix}
   Q & c \\
   c^* & 1
   \end{bmatrix} \succeq 0
   \]
2. \( \operatorname{Tr}(Q) = 1 \)
3. sums along superdiagonals vanish: \( \sum_{i=1}^{n-j} Q_{i,i+j} = 0 \text{ for } 1 \leq j \leq n - 1 \)
SDP formulation

Dual as an SDP

maximize \( \Re \langle y, c \rangle \)
subject to
\[
\begin{bmatrix}
Q & c \\
c^* & 1
\end{bmatrix} \succeq 0
\]
\[
\sum_{i=1}^{n-j} Q_{i,i+j} = \delta_j \quad 0 \leq j \leq n - 1
\]

Dual solution \( c \): coeffs. of low-pass trig. polynomial \( \sum_k c_k e^{i2\pi kt} \) interpolating the sign of the primal solution
SDP formulation

Dual as an SDP

maximize $\text{Re}\langle y, c \rangle$ subject to

$$
\begin{bmatrix}
Q & c \\
\bar{c}^* & 1
\end{bmatrix} \succeq 0
$$

$$
\sum_{i=1}^{n-j} Q_{i,i+j} = \delta_j \quad 0 \leq j \leq n - 1
$$

Dual solution $c$: coeffs. of low-pass trig. polynomial $\sum_k c_k e^{i2\pi kt}$ interpolating the sign of the primal solution

To recover spike locations

(1) Solve dual

(2) Check when polynomial takes on magnitude 1
With noise

\[ y = F_n x + \text{noise} \]

\[
\min \quad \frac{1}{2} \| y - F_n \tilde{x} \|_2^2 + \lambda \| \tilde{x} \|_{TV}
\]

- Also an SDP
- Theory: C. and Fernandez Granda (’12)
Noisy example

SNR: 14 dB
Noisy example

SNR: 14 dB
Noisy example

SNR: 14 dB
Noisy example

Average localization error: $6.54 \times 10^{-4}$
Summary

- Three important problems with missing data
  - Phase retrieval
  - Matrix completion/RPCA
  - Super-resolution

- Three simple and model-free recovery procedures via convex programming

- Three near-perfect solutions
Apologies: things I have not talked about

- Algorithms
- Applications
- Avalanche of related works
A small sample of papers I have greatly enjoyed

- Phase retrieval
  - Netrapalli, Jain, Sanghavi, *Phase retrieval using alternating minimization* ('13)
  - Waldspurger, d’Aspremont, Mallat, *Phase recovery, MaxCut and complex semidefinite programming* ('12)

- Robust PCA
  - Gross, *Recovering low-rank matrices from few coefficients in any basis* ('09)
  - Hsu, Kakade and Zhang, *Robust matrix decomposition with outliers* ('11)

- Super-resolution
  - Kahane, *Analyse et synthèse harmoniques* ('11)
  - Slepian, *Prolate spheroidal wave functions, Fourier analysis, and uncertainty. V - The discrete case* ('78)
Nuclear norm and spectral norms are dual: $\|X\|_* = \text{val}(P)$

\[ (P) \quad \text{maximize} \quad \langle U, X \rangle \quad \text{subject to} \quad \|U\| \leq 1 \quad \Leftrightarrow \quad \text{maximize} \quad \langle U, X \rangle \quad \text{subject to} \quad \begin{bmatrix} I & U \\ U^* & I \end{bmatrix} \preceq 0 \]
General SDP formulation

Nuclear norm and spectral norms are dual: \( \|X\|_* = \text{val}(P) \)

\[
\begin{align*}
(P) \quad & \text{maximize} & \langle U, X \rangle \\
& \text{subject to} & \|U\| \leq 1 \\
\end{align*}
\]

\[
\begin{align*}
(D) \quad & \text{minimize} & .5(\text{Tr}(W_1) + \text{Tr}(W_2)) \\
& \text{subject to} & \begin{bmatrix} W_1 & X \\ X^* & W_2 \end{bmatrix} \preceq 0 \\
\end{align*}
\]

Optimization variables: \( W_1 \in \mathbb{R}^{n_1 \times n_1}, W_2 \in \mathbb{R}^{n_2 \times n_2} \)

The super-resolution factor

Have data at resolution $\lambda_{lo}$

Wish resolution $\lambda_{hi}$

Super-resolution factor

$$SRF = \frac{\lambda_{lo}}{\lambda_{hi}}$$
The super-resolution factor (SRF): frequency viewpoint

- Observe spectrum up to $f_{lo}$
- Wish to extrapolate up to $f_{hi}$

Super-resolution factor

$$SRF = \frac{f_{hi}}{f_{lo}}$$
With noise

\[ y = \mathcal{F}_n x + \text{noise} \]

\[ \mathcal{F}_n x = \int_0^1 e^{-i2\pi kt} x(dt) \]

\[ |k| \leq f_\text{lo} \]

At ‘native’ resolution

\[ \| (\hat{x} - x) \ast \varphi_{\lambda_\text{lo}} \|_{\text{TV}} \lesssim \text{noise level} \]
With noise

\[ y = F_n x + \text{noise} \]
\[ F_n x = \int_0^1 e^{-i2\pi kt} x(dt) \]
\[ |k| \leq f_{lo} \]

At ‘finer’ resolution \( \lambda_{hi} = \lambda_{lo}/\text{SRF} \), convex programming achieves

\[ \| (\hat{x} - x) \ast \varphi_{\lambda_{hi}} \|_{TV} \leq \text{SRF}^2 \times \text{noise level} \]
With noise

\[ y = \mathcal{F}_n x + \text{noise} \]
\[ \mathcal{F}_n x = \int_0^1 e^{-i2\pi kt} x(dt) \]
\[ |k| \leq f_{lo} \]

At ‘finer’ resolution \( \lambda_{hi} = \lambda_{lo}/\text{SRF} \), convex programming achieves

\[ \| (\hat{x} - x) \ast \varphi_{\lambda_{hi}} \|_{TV} \lesssim \text{SRF}^2 \times \text{noise level} \]

Modulus of continuity studies for super-resolution: Donoho (’92)