

Compressed sensing and low-rank optimization

Roman Schutski

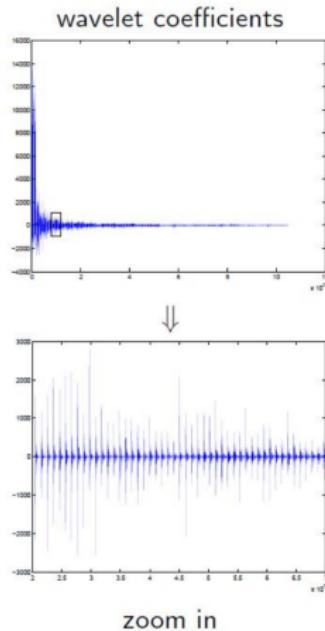
Skoltech

November 27, 2018

Natural signals are sparse



1 megapixel image



Implication

We can discard most of small coefficients without loosing perceptual quality

Compressing data

- ▶ Calculate 1,000 000 wavelet/cosine/other coefficients of the image
- ▶ Drop all but 25,000 largest coefficients
- ▶ Invert the transformation



1 megapixel image



25k term approximation

Usual path: collect full → transform → shrink

Idea

What if we try to work with a limited subset of data?

Underdetermined problems

- ▶ Have signal $x \in \mathbf{R}^n$, sample a linear combination $A \in \mathbf{R}^{m \times n}$ of entries into $b \in \mathbf{R}^m$, where $m \ll n$

$$\begin{bmatrix} b \\ = \\ \end{bmatrix} \quad A \quad \begin{bmatrix} x \\ \end{bmatrix}$$

- ▶ In general it is not possible to solve for x by the fundamental theorem of algebra

Special structure

$$\begin{bmatrix} b \\ A \end{bmatrix} = \begin{bmatrix} * \\ * \\ * \\ x \end{bmatrix}$$

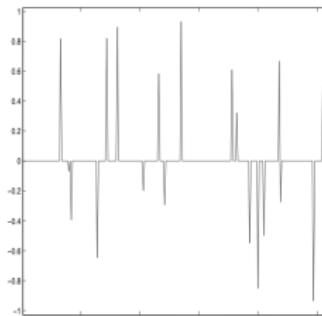
If unknowns are assumed

- ▶ Sparse
- ▶ Low-rank

this *may* be possible by convex optimization

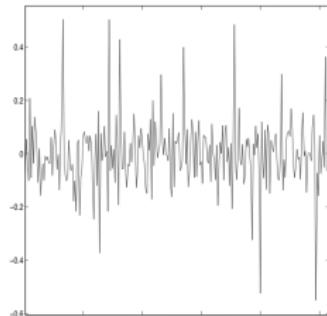
Example

- Given $n = 256$, $m = 128$
- $A = \text{numpy.random}(m, n)$
 $x = \text{scipy.sparse.random}(n, 1, \text{density}=0.1)$
 $b = A \cdot \text{dot}(x)$

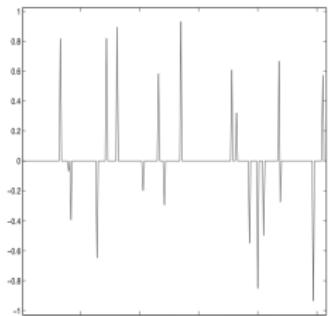


$$\begin{cases} \min_x \|x\|_0 \\ \text{s.t. } Ax = b \end{cases}$$

(a) exact



$$\begin{cases} \min_x \|x\|_2 \\ \text{s.t. } Ax = b \end{cases}$$

(b) ℓ_2 -minimization

$$\begin{cases} \min_x \|x\|_1 \\ \text{s.t. } Ax = b \end{cases}$$

(c) ℓ_1 -minimization

Linear programming formulation

L_0 norm

- ▶ $\|x\|_0 = \text{number of nonzero elements in } x$
- ▶ Not convex, NP-hard to find a minimum

L_1 norm

- ▶ $\|x\|_1 = \sum_i |x_i|$
- ▶ Convex

$$\begin{aligned} & \text{minimize} && \sum_i |x_i| \\ & \text{subject to} && Ax = b \end{aligned}$$

is equivalent to

$$\begin{aligned} & \text{minimize} && \sum_i t_i \\ & \text{subject to} && Ax = b \\ & && -t_i \leq x_i \leq t_i \end{aligned}$$

with variables $x, t \in \mathbb{R}^n$

$$x^* \text{ solution} \iff (x^*, t^* = |x^*|) \text{ solution}$$

When does this work?

- ▶ The field is rapidly evolving and there is no "clean" theory yet
- ▶ The requirement $A_{ij} \in \mathcal{N}(0, 1)$ is important
- ▶ I will highlight only some aspects of the following:
 - ▶ How many measurements needed to get solution?
 - ▶ What if x is not completely sparse?
 - ▶ For what matrices A sparse problem is solvable?
- ▶ Not covered:
 - ▶ Uniqueness of solutions/properties of L_1 -norm relaxation/many other aspects

How many measurements are needed

Result 1 (Candes, Romberg, Tao '06; Donoho, '06)

Suppose $x \in \mathbf{R}^n$ and $\|x\|_0 = s$ (x is s -sparse). Let $A \in \mathbf{R}^{n \times m}$, $A_{ij} \in \mathcal{N}(0, 1) \quad \forall i, j$, where

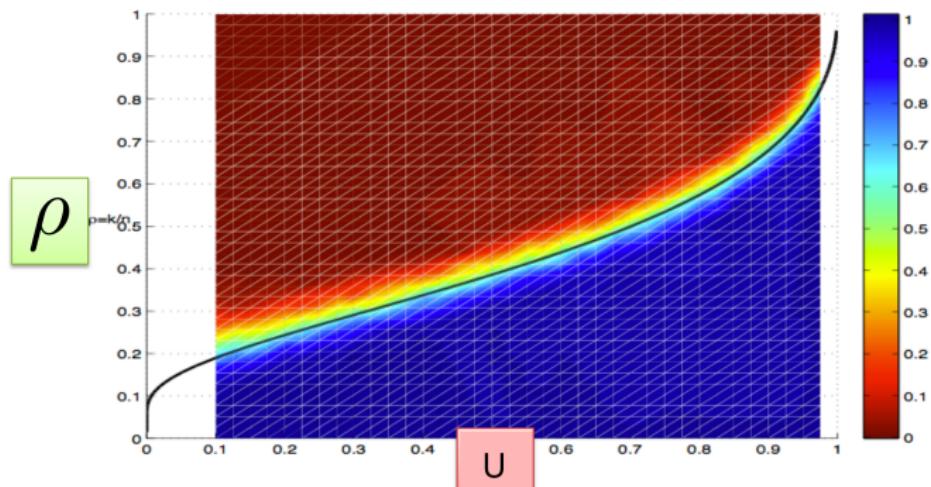
$$m \geq s \cdot \log \frac{n}{s}$$

With high probability over the choice of A the solution of $Ax = b$ can be recovered.

Let us define $\frac{m}{n} = u$ (undersampling fraction) and $\frac{s}{n} = \rho$ (sparsity fraction).

Phase transitions

- Under the curve is almost complete recovery, over the curve is almost complete failure!



Building phase transition curve

For a particular matrix A and reconstruction algorithm:

- ▶ Choose ρ, u
- ▶ Monte Carlo simulation of many problem instances
- ▶ Count ratio of successes vs. total problem instances

What if x is not exactly sparse?

Result 2 (C. Romberg and Tao)

Suppose we have a setup of the previous theorem, x is not exactly sparse. If

$$m \geq s \cdot \log \frac{n}{s}$$

and \hat{x} is a s -sparse solution, then

$$\|\hat{x} - x\|_2 \leq \|x - x_s\|_1$$

where x_s contains s largest values of x

Corollary

We can look for sparse solutions when x contains noise or is only approximately sparse!

Corollary: Sparse approximation for noisy data

- ▶ x is s -sparse, inaccurate measurements: z error term (stochastic or deterministic)

$$b = Ax + z, \text{ with } \|z\|_{\ell_2} \leq \epsilon$$

- ▶ Recovery via the LASSO:

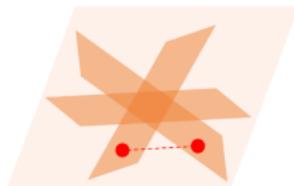
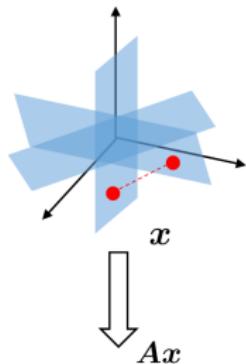
$$\min \|\hat{x}\|_{\ell_1} \text{ s. t. } \|A\hat{x} - b\|_{\ell_2} \leq \epsilon$$

- ▶ By the Result 2, the error of the final solution is:

$$\|\hat{x} - x\|_{\ell_2} \lesssim \frac{\|x - x_s\|_{\ell_1}}{\sqrt{s}} + \epsilon = \text{approx.error} + \text{measurement error}$$

What sampling matrices A allow sparse solutions?

- ▶ Restricted Isometry Property: suppose x is s -sparse. Then, the smallest possible ϵ_{2s} , s.t.
$$(1 - \epsilon) \|x_1 - x_2\|_F \leq \|Ax_1 - Ax_2\|_2^2 \leq (1 + \epsilon) \|x_1 - x_2\|_F$$
 for all x_1, x_2 is the $2s$ -isometry constant of A .
- ▶ With a picture equivalently: $(1 - \epsilon) \leq \frac{\|Ax_1 - Ax_2\|_2^2}{\|x_1 - x_2\|_F} \leq (1 + \epsilon)$



Restricted Isometry Property

- ▶ RIP property is a sufficient condition for sparse algorithms convergence/solution uniqueness
- ▶ If $\epsilon_{2s} < 1$ then L_0 minimization problem has a unique sparse solution
- ▶ if $\epsilon_{2s} < (\sqrt{2} - 1) = 0.414$ then L_1 relaxed problem has a **the same** unique solution.
- ▶ Examples of "good" matrices A :
 - ▶ Random $m \times n$ matrices with iid elements if $m \gg s \cdot \log(n/s)$
 - ▶ Random $m \times n$ partial DFT matrices if $m \gg s \cdot \log^4(n)$
 - ▶ etc

Extending the sparsity concept

- » **Sparse Vectors:** linear combination of standard basis

$$\mathbf{x} = \sum_i c_i \mathbf{e}_i$$

- » **Low Rank Matrices:** linear combination of rank-1 matrices

$$\mathbf{X} = \sum_i c_i \mathbf{u}_i \mathbf{v}_i^T$$

- ▶ **From sparsity to low rank matrices:** the problem is still about finding a sparse solution, but in terms of singular values

$$X \longrightarrow \sigma(X)$$

Netflix problem

- Predict ratings of movies across all users

$$R \in \mathbb{R}^{m \times n}$$

Movies

$$\begin{bmatrix} 2 & 3 & ? & ? & 5 & ? \\ 1 & ? & ? & 4 & ? & 3 \\ ? & ? & 3 & 2 & ? & 5 \\ 4 & ? & 3 & ? & 2 & 4 \end{bmatrix}$$

Users

- Only know R_{ij} for $i, j \in \Omega$
- Don't have ratings of every movie from every user

Low rank matrix decomposition

- ▶ Can “explain” a movie rating by a small (k) number of features
 - ▶ Actors, genre, storyline, length, year, ...
- ▶ Each user has a preference for the features

$$R_{ij} = u_i^T v_j$$

Diagram illustrating the decomposition:

On the left, a vertical vector u_i is shown as a stack of red rectangles. Below it, the text "User's interest in each feature" is written.

In the center, a horizontal vector v_j is shown as a stack of blue rectangles. Below it, the text "Feature vector" is written.

Between the two vectors is an equals sign (=).

To the right of the equals sign is a single orange square.

Below the orange square, the text "User i's rating of movie j" is written.

Low rank matrix decomposition

- ▶ Can “explain” a movie rating by a small (k) number of features
 - ▶ Actors, genre, storyline, length, year, ...
- ▶ Each user has a preference for the features
- ▶ Matrix R is low rank, with rank $k \ll m, k \ll n$

$$R = UV^T$$

$$U \in \mathbb{R}^{m \times k}$$
$$V^T \in \mathbb{R}^{k \times n}$$

Low rank matrix decomposition

- Given that the true R is low rank, find a matrix X that is low rank and agrees with R at the observed entries:

$$\underset{X}{\text{minimize}} \quad \text{rank } X$$

$$\text{subject to} \quad \mathcal{A}(X) = y$$

$$(X_{ij} = R_{ij} \quad \forall(i, j) \in \Omega)$$

- Rank(X) is not a convex function!

Low rank matrix decomposition

- Given that the true R is low rank, find a matrix X that is low rank and agrees with R at the observed entries:

$$\underset{X}{\text{minimize}} \quad \text{rank } X$$

$$\text{subject to} \quad \mathcal{A}(X) = y$$

$$(X_{ij} = R_{ij} \quad \forall(i, j) \in \Omega)$$

- Nuclear norm instead of the rank function (Recht, Fazel, Parrilo):

$$||X||_* = \sum_i \sigma_i \quad (\text{sum of singular values})$$

Low rank algorithm intuition

- ▶ How can we solve the low rank matrix completion problem?
- ▶ **Intuition:**
 - ▶ A low rank matrix has a small number of non-zero singular values
 - ▶ We see a linear mixture of these singular values (through SVD)
 - ▶ Apply soft-thresholding iteratively on the singular values of X
- ▶ **Projection onto convex sets:**
 - ▶ Take the SVD: $X = P\Sigma Q^T$ - not low rank
 - ▶ Soft Threshold: $\hat{\Sigma} = S_\lambda(\Sigma)$
 - ▶ Form new matrix: $\hat{X} = P\hat{\Sigma}Q^T$ - low rank but inconsistent with R_{ij}
 - ▶ Enforce constraints (replace entries): $\hat{X}_{ij} = R_{ij}$

When does this work

- ▶ When the rows/columns of R are incoherent. More specifically, if $\forall X$, $\text{rank}(X) \leq r$, $\exists \epsilon < 0.414$ s.t.:

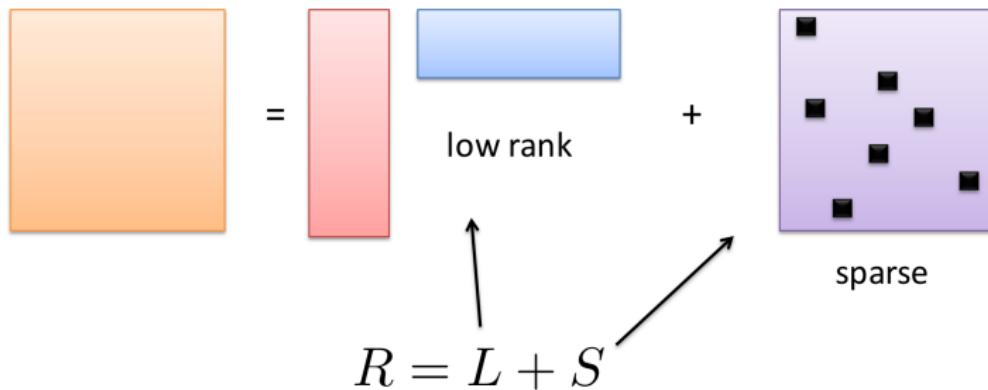
$$(1 - \epsilon)\|X\|_F^2 \leq \|\mathcal{A}(X)\|_2^2 \leq (1 + \epsilon)\|X\|_F^2$$

The latter condition is Restricted Isometry Property, satisfied if entries of R are at random positions

- ▶ **Theorem:**
 - ▶ If X is rank- r and RIP holds for \mathcal{A} and $Y = \mathcal{A}(X)$
 - ▶ Then "Projection onto convex sets" converges to the optimum of the problem
- ▶ **Uniqueness:**
 - ▶ There are multiple solutions to the low-rank problem, but only one is sparse

Low-rank + sparse decomposition

- ▶ Low-rank matrix with sparse errors



- ▶ Given R , find L and S exactly
- ▶ Not a well-posed problem in general

Low-rank + sparse decomposition

- ▶ Bad cases:
 - ▶ L is low-rank **and** sparse
 - ▶ rows/columns of S are coherent

$$L = \begin{matrix} & \\ & \blacksquare \\ & \end{matrix}$$

No hope of separating from S

$$S = \begin{matrix} & \\ & \text{---} \\ & \end{matrix}$$

No hope of recovering first row of L

- ▶ Exact conditions are given in Candès E., Li X., Ma Y., Wright J. *Robust principal component analysis?*

Low-rank + sparse decomposition

- ▶ Under the conditions presented in (Candès, Li, Ma, Wright) the following convex problem has a unique solution:

$$\underset{L,S}{\text{minimize}} \quad \|L\|_* + \lambda \|S\|_1$$

$$\text{subject to} \quad R = L + S$$

Low-rank + sparse decomposition

- ▶ Application: background separation
 - ▶ Background is slowly changing between frames → low rank
 - ▶ Fast changing components are rare → sparse

Input Video

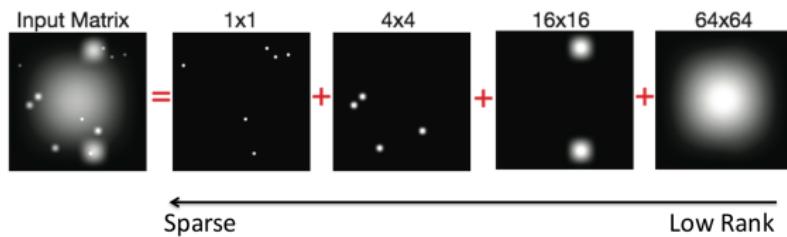
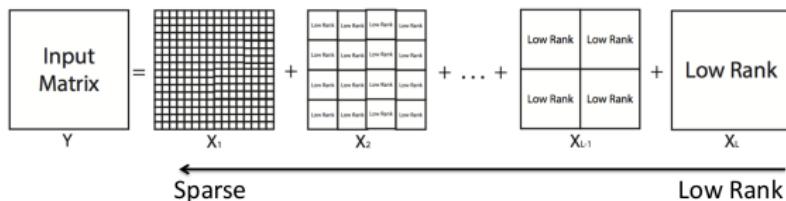


**Low Rank
+
Sparse**



Multiscale low-rank + sparse

- ▶ Sum of matrices with increasing scales of correlation (aka wavelets)
- ▶ Do decomposition at all scales



- ▶ Full details in: F. Ong, M Lustig, *Beyond low rank+ sparse: Multiscale low rank matrix decomposition*, 2016

Multiscale low-rank + sparse

- ▶ Algorithm idea:

$$\begin{aligned} & \underset{X_i}{\text{minimize}} && \sum_{i=0}^{L-1} \lambda_i \|X_i\|_{(i)} \\ & \text{subject to} && Y = \sum_{i=0}^{L-1} X_i \end{aligned}$$

- ▶ "Projection onto convex sets" for blocks:
 - ▶ Enforce block low rank for each X_i (Block-wise SVD + iterative soft thresholding)
 - ▶ Enforce data consistency

Example of multiscale low-rank + sparse

Input Video



**Multi-scale
Low Rank**

