

# MA3113: Topics in Mathematical Image Processing I

## Sparse Representation and Dictionary Learning



Suh-Yuh Yang (楊肅煜)

Department of Mathematics, National Central University  
Jhongli District, Taoyuan City 320317, Taiwan

First version: April 18, 2022/Last updated: April 15, 2024

# Outlines

---

- ① Sparse representation and dictionary learning
  - Sparse representation (SR) problem
  - Alternating direction method of multipliers (ADMM)
  - Application to signal denoising
  - Sparse dictionary learning (SDL) problem
  - Solving SDL problem
- ② Convolutional sparse representation and dictionary learning
  - ADMM for solving convolutional SR problem
  - Convolutional SDL problem
  - Solving convolutional SDL problem

## Part I

# Sparse Representation and Dictionary Learning

## Sparse representation problem

---

**Terms:** Sparse Representation (稀疏表現)/Sparse Coding (稀疏編碼)

**SR problem:** *Given a signal vector  $\mathbf{x} \in \mathbb{R}^m$  and a dictionary matrix  $\mathbf{D} \in \mathbb{R}^{m \times n}$ , we seek a sparse coefficient vector  $\mathbf{z}^* \in \mathbb{R}^n$  such that*

$$\mathbf{z}^* = \arg \min_{\mathbf{z}} \left( \frac{1}{2} \|\mathbf{x} - \mathbf{D}\mathbf{z}\|_2^2 + \lambda \|\mathbf{z}\|_0 \right),$$

*where  $\lambda > 0$  is a penalty parameter and  $\|\mathbf{z}\|_0$  counts the number of nonzero components of  $\mathbf{z}$ .*

### Remarks:

- In the matrix-vector multiplication  $\mathbf{D}\mathbf{z}$ , the components of  $\mathbf{z}$  are the coefficients with respect to columns (also called *atoms*) of  $\mathbf{D}$ .
- We call  $\|\mathbf{z}\|_0$  the  $\ell^0$  norm of  $\mathbf{z}$ , even though  $\ell^0$  is *not* really a norm, since the *homogeneity property* fails,  $\|\alpha\mathbf{z}\|_0 \neq |\alpha| \|\mathbf{z}\|_0$ .
- It is inefficient to compute  $\|\mathbf{z}\|_0$  directly when  $n$  is large. In practice, we will use the  $\ell^1$  norm instead of the  $\ell^0$  norm.

## Two dual $\ell^0$ minimization problems

---

In [Sharon-Wright-Ma 2007], they studied the following two *dual*  $\ell^0$  minimization problems:

- **Sparse error correction (SEC):** *Given  $\mathbf{0} \neq \mathbf{y} \in \mathbb{R}^n$  and  $A \in \mathbb{R}^{n \times p}$  with  $n > p$  and  $\text{rank}(A) = p$ , we seek  $\mathbf{w}^* \in \mathbb{R}^p$  such that*

$$\mathbf{w}^* = \arg \min_w \|\mathbf{y} - A\mathbf{w}\|_0. \quad (1)$$

- **Sparse signal reconstruction (SSR):** *Given  $D \in \mathbb{R}^{m \times n}$  with  $m < n$  and  $\mathbf{0} \neq \mathbf{x} \in C(D)$  the column space of  $D$ , we seek  $\mathbf{z}^* \in \mathbb{R}^n$  such that*

$$\mathbf{z}^* = \arg \min_z \|\mathbf{z}\|_0 \quad \text{subject to} \quad \mathbf{x} = D\mathbf{z}. \quad (2)$$

Note that (1) is a *decoding problem*, while (2) is a *sparse representation problem*. These two problems are *dual* in the sense that we can convert one problem to the other, see page 8 below.

# Existence and uniqueness of solution

## 1 Existence:

- Existence of  $w^*$ : If  $\exists w \in \mathbb{R}^p$  s.t.  $\|y - Aw\|_0 = 0$ , then  $w^* = w$ . Otherwise, define

$$\mathcal{S} := \{k \in \mathbb{N} : \exists w \in \mathbb{R}^p \text{ s.t. } \|y - Aw\|_0 = k\}.$$

Then  $\emptyset \neq \mathcal{S} \subseteq \mathbb{N}$ . By the well-ordering principle,  $\exists k_0 \in \mathcal{S}$  the minimum of  $\mathcal{S}$ . i.e.,  $\exists w^*$  such that  $w^* = \arg \min_w \|y - Aw\|_0$ .

- Existence of  $z^*$ : It can be shown in a similar way!

## 2 Uniqueness: It will generally be true that these two dual problems have a unique solution if

- $\exists w_0$  such that the error  $e := y - Aw_0$  is sparse enough.  
*The proof needs the “null space property” and the “restricted isometry property.”*
- $\exists z_0$  sparse enough such that  $x = Dz_0$ .  
*e.g., if any set of  $2T$  columns of  $D$  are linearly independent, then any  $z_0 \in \mathbb{R}^n$  with  $\|z_0\|_0 \leq T$  such that  $Dz_0 = x$  is the unique solution to SSR problem (2).*

## Why we require matrix $A$ full rank $p$ in the SEC problem?

---

Note that  $A$  is of size  $n \times p$  and  $n > p$ .

Suppose that  $A$  is not full rank  $p$ . Then  $\text{rank}(A) < p$ .

Since  $\dim N(A) + \text{rank}(A) = p$ , we have  $\dim N(A) > 0$ .

Thus, nullspace  $N(A) \neq \{\mathbf{0}\}$  and  $\exists \tilde{\mathbf{w}} \neq \mathbf{0}$  such that  $A\tilde{\mathbf{w}} = \mathbf{0}$ .

If  $\mathbf{w}^*$  is a solution of the SEC problem, then

$$\|\mathbf{y} - A(\mathbf{w}^* + \tilde{\mathbf{w}})\|_0 = \|\mathbf{y} - A\mathbf{w}^*\|_0.$$

Hence,  $\mathbf{w}^* + \tilde{\mathbf{w}}$  is also a solution of the SEC problem.

*Therefore, in order to ensure the uniqueness, we require  $A$  full rank  $p$ .*

## How to convert problem (2) to problem (1)?

- The decoding problem (1) can be converted to the sparse representation problem (2). [Candès *et al.* 2005, IEEE Symposium on FOCS]
- **Converting (2) to (1):** Let  $p = n - \text{rank}(\mathbf{D}) > 0$  and  $\mathbf{A}$  be a full-rank  $n \times p$  matrix whose columns span the nullspace of  $\mathbf{D}$ , i.e.,  $\mathbf{D}\mathbf{A} = \mathbf{0}$ . Find any  $\mathbf{y} \in \mathbb{R}^n$  so that  $\mathbf{D}\mathbf{y} = \mathbf{x}$  and define  $f(\mathbf{w}) = \mathbf{y} - \mathbf{A}\mathbf{w}$ . Then

$$\underbrace{\arg \min_{\mathbf{z} \ \& \ \mathbf{D}\mathbf{z}=\mathbf{x}} \|\mathbf{z}\|_0}_{\mathbf{z}^*} = f\left(\underbrace{\arg \min_{\mathbf{w}} \|\mathbf{y} - \mathbf{A}\mathbf{w}\|_0}_{f(\mathbf{w}^*)}\right). \quad (3)$$

*Proof:* First, note that for all  $\mathbf{w} \in \mathbb{R}^p$ , we have

$$Df(\mathbf{w}) = D(\mathbf{y} - \mathbf{A}\mathbf{w}) = D\mathbf{y} - D\mathbf{A}\mathbf{w} = D\mathbf{y} = \mathbf{x} \implies \|f(\mathbf{w})\|_0 \geq \|\mathbf{z}^*\|_0.$$

Claim:  $\exists \tilde{\mathbf{w}} \in \mathbb{R}^p$  such that  $f(\tilde{\mathbf{w}}) = \mathbf{y} - \mathbf{A}\tilde{\mathbf{w}} = \mathbf{z}^*$ .

$$\because D\mathbf{z}^* = \mathbf{x} \text{ and } D(\mathbf{y} - \mathbf{A}\mathbf{w}) = \mathbf{x}, \forall \mathbf{w} \implies D(-\mathbf{z}^* + \mathbf{y} - \mathbf{A}\mathbf{w}) = \mathbf{0}$$

$$\therefore \exists \tilde{\mathbf{w}} \text{ such that } \mathbf{A}\tilde{\mathbf{w}} = -\mathbf{z}^* + \mathbf{y} - \mathbf{A}\mathbf{w} \implies \mathbf{z}^* = \mathbf{y} - \mathbf{A}(\mathbf{w} + \tilde{\mathbf{w}}) := f(\tilde{\mathbf{w}})$$

Claim:  $\tilde{\mathbf{w}} = \mathbf{w}^* := \arg \min_{\mathbf{w}} \|\mathbf{y} - \mathbf{A}\mathbf{w}\|_0$ , and then  $f(\mathbf{w}^*) = \mathbf{z}^*$ .

$$\because \|f(\mathbf{w}^*)\|_0 \leq \|f(\tilde{\mathbf{w}})\|_0 = \|\mathbf{z}^*\|_0 \leq \|f(\mathbf{w}^*)\|_0 \implies \|f(\mathbf{w}^*)\|_0 = \|f(\tilde{\mathbf{w}})\|_0$$

By the uniqueness of  $\mathbf{w}^*$ , we obtain  $\tilde{\mathbf{w}} = \mathbf{w}^*$  and then  $f(\mathbf{w}^*) = \mathbf{z}^*$ .

## The $\ell^1$ - $\ell^0$ equivalence problem

- In general, the  $\ell^0$  minimizations (1) and (2) are *NP-hard problems*:

$$\mathbf{w}^* = \arg \min_w \|\mathbf{y} - A\mathbf{w}\|_0, \quad (1)$$

$$\mathbf{z}^* = \arg \min_z \|\mathbf{z}\|_0 \quad \text{subject to} \quad \mathbf{x} = D\mathbf{z}. \quad (2)$$

- The equivalence between  $\ell^0$  and  $\ell^1$  minimizations is conditional.

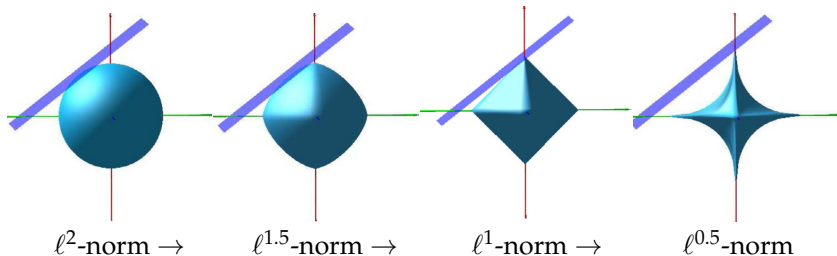
David L. Donoho, *For most large underdetermined systems of linear equations the minimal  $\ell_1$ -norm solution is also the sparsest solution*, CPAM, 59 (2006), pp. 797-829.

*If the error  $\mathbf{e} := \mathbf{y} - A\mathbf{w}^*$  and the solution  $\mathbf{z}^*$  is sufficiently sparse, then the solutions to (1) and (2) are the same as (4) and (5), respectively.*

$$\mathbf{w}^* = \arg \min_w \|\mathbf{y} - A\mathbf{w}\|_1, \quad (4)$$

$$\mathbf{z}^* = \arg \min_z \|\mathbf{z}\|_1 \quad \text{subject to} \quad \mathbf{x} = D\mathbf{z}. \quad (5)$$

## 3-D ball in different $\ell^r$ norms and the constraint $Dz = x$



*3-D ball in the different  $\ell^r$  norms for  $r = 2, 1.5, 1, 0.5$*

$$z^* = \arg \min_z \|z\|_1 \quad \text{subject to} \quad \underbrace{x}_{\text{given}} = D \underbrace{z}_{\text{many}} \quad (5)$$

## The sparse representation problem

- We have introduced some ideas about the  $\ell^1$ - $\ell^0$  equivalence. In what follows, we don't consider the original SR problem. We consider the following  $\ell^1$  minimization problem instead:

**SR problem:** *Given a signal vector  $\mathbf{x} \in \mathbb{R}^m$  and a dictionary matrix  $\mathbf{D} \in \mathbb{R}^{m \times n}$ , we seek a coefficient vector  $\mathbf{z}^* \in \mathbb{R}^n$  such that*

$$\mathbf{z}^* = \arg \min_z \left( \frac{1}{2} \|\mathbf{x} - \mathbf{D}\mathbf{z}\|_2^2 + \lambda \|\mathbf{z}\|_1 \right), \quad \lambda > 0. \quad (\star)$$

The existence (and uniqueness) of solution of the SR problem  $(\star)$  can be ensured because matrix  $\mathbf{D}^\top \mathbf{D}$  is symmetric (+ *positive definite*) and the second term  $\lambda \|\cdot\|_1$  is a *convex function*.

- Problem  $(\star)$  is also a regression analysis method in statistics and machine learning. It is the so-called *least absolute shrinkage and selection operator (LASSO)*.

R. J. Tibshirani, The lasso problem and uniqueness, *Electronic Journal of Statistics*, 7 (2013), pp. 1456-1490  $\oplus$  A. Ali, 13 (2019), pp. 2307-2347.

## Alternating direction method of multipliers (ADMM)

---

We will use the “*Alternating Direction Method of Multipliers*” to solve the above  $\ell^1$ -norm SR problem.

- ADMM is an iterative scheme for solving the following equality constrained convex optimization problems:

$$\min_z f(z) \quad \text{subject to} \quad Az = b.$$

- ADMM consists of three steps:
  1. adding an auxiliary variable  $\mathbf{y}$  and a dual variable (multipliers)  $\mathbf{v}$  and then scaled as  $\mathbf{u}$
  2. separating the new cost function into a sum of  $f(z)$  and  $g(\mathbf{y})$
  3. using an iterative method to solve the problem
- Then the optimization problem can be re-posed as

$$\min_{z, \mathbf{y}} (f(z) + g(\mathbf{y})) \quad \text{subject to} \quad Az + B\mathbf{y} = c.$$

## Derivation of the ADMM: augmented Lagrangian

First, we formulate the *augmented Lagrangian*

$$L_\rho(\mathbf{z}, \mathbf{y}, \mathbf{v}) := f(\mathbf{z}) + g(\mathbf{y}) + \underbrace{\mathbf{v}^\top}_{\text{multipliers}} (\mathbf{A}\mathbf{z} + \mathbf{B}\mathbf{y} - \mathbf{c}) + \underbrace{\frac{\rho}{2} \|\mathbf{A}\mathbf{z} + \mathbf{B}\mathbf{y} - \mathbf{c}\|_2^2}_{\text{penalty term}}$$

where  $\rho > 0$  is the penalty parameter. Then the iterative scheme of the *augmented Lagrangian method* (ALM) is given by

$$\begin{aligned}(\mathbf{z}^{(i+1)}, \mathbf{y}^{(i+1)}) &= \arg \min_{\mathbf{z}} L_\rho(\mathbf{z}, \mathbf{y}, \mathbf{v}^{(i)}), \\ \mathbf{v}^{(i+1)} &= \mathbf{v}^{(i)} + \rho(\mathbf{A}\mathbf{z}^{(i+1)} + \mathbf{B}\mathbf{y}^{(i+1)} - \mathbf{c}).\end{aligned}$$

In ADMM,  $\mathbf{z}$  and  $\mathbf{y}$  are updated in an alternating or sequential fashion, which accounts for the term *alternating direction*.

$$\begin{aligned}\mathbf{z}^{(i+1)} &= \arg \min_{\mathbf{z}} L_\rho(\mathbf{z}, \mathbf{y}^{(i)}, \mathbf{v}^{(i)}), \\ \mathbf{y}^{(i+1)} &= \arg \min_{\mathbf{y}} L_\rho(\mathbf{z}^{(i+1)}, \mathbf{y}, \mathbf{v}^{(i)}), \\ \mathbf{v}^{(i+1)} &= \mathbf{v}^{(i)} + \rho(\mathbf{A}\mathbf{z}^{(i+1)} + \mathbf{B}\mathbf{y}^{(i+1)} - \mathbf{c}).\end{aligned}$$

## Scaled form of the augmented Lagrangian

---

The ADMM can be written in a slightly different form, which is often more convenient, by combining the linear and quadratic terms in the augmented Lagrangian and scaling the dual variable (multipliers)  $v$ .

Define the residual  $r := Az + By - c$ . Then

$$\begin{aligned} v^\top (Az + By - c) + \frac{\rho}{2} \|Az + By - c\|_2^2 \\ = v^\top r + \frac{\rho}{2} \|r\|_2^2 = \frac{\rho}{2} \|r + \frac{1}{\rho} v\|_2^2 - \frac{1}{2\rho} \|v\|_2^2. \end{aligned}$$

Set  $u = \frac{1}{\rho} v$ . Then  $L_\rho(z, y, v) = L_\rho(z, y, u)$ , and

$$L_\rho(z, y, u) = f(z) + g(y) + \frac{\rho}{2} \|Az + By - c + u\|_2^2 - \frac{\rho}{2} \|u\|_2^2.$$

## ADMM: scaled form

---

The ADMM in the scaled form is given by

$$\begin{aligned} \mathbf{z}^{(i+1)} &= \arg \min_{\mathbf{z}} \left( f(\mathbf{z}) + g(\mathbf{y}^{(i)}) + \frac{\rho}{2} \|\mathbf{Az} + \mathbf{By}^{(i)} - \mathbf{c} + \mathbf{u}^{(i)}\|_2^2 - \frac{\rho}{2} \|\mathbf{u}^{(i)}\|_2^2 \right), \\ \mathbf{y}^{(i+1)} &= \arg \min_{\mathbf{y}} \left( f(\mathbf{z}^{(i+1)}) + g(\mathbf{y}) + \frac{\rho}{2} \|\mathbf{Az}^{(i+1)} + \mathbf{By} - \mathbf{c} + \mathbf{u}^{(i)}\|_2^2 - \frac{\rho}{2} \|\mathbf{u}^{(i)}\|_2^2 \right), \\ \mathbf{u}^{(i+1)} &= \mathbf{u}^{(i)} + \mathbf{Az}^{(i+1)} + \mathbf{By}^{(i+1)} - \mathbf{c}, \end{aligned}$$

where  $\rho > 0$  is the *penalty parameter* which is related to the convergent rate of the iterations.

*Note that the terms in blue can be omitted in practical computations!*

**Reference:** S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, Distributed optimization and statistical learning via the ADMM, *Foundations and Trends<sup>®</sup> in Machine Learning*, 3 (2010), pp. 1-122.

## ADMM for the $\ell^1$ -norm SR problem

- For the  $\ell^1$ -norm SR problem,

$$\mathbf{z}^* = \arg \min_z \left( \frac{1}{2} \|\mathbf{x} - \mathbf{Dz}\|_2^2 + \lambda \|\mathbf{z}\|_1 \right), \quad \lambda > 0, \quad (\star)$$

we set

$$f(\mathbf{z}) := \frac{1}{2} \|\mathbf{x} - \mathbf{Dz}\|_2^2, \quad g(\mathbf{y}) := \lambda \|\mathbf{y}\|_1, \quad \mathbf{Az} + \mathbf{By} = \mathbf{c} \Leftrightarrow \mathbf{z} - \mathbf{y} = \mathbf{0}.$$

- The ADMM for the  $\ell^1$ -norm SR problem is given by

$$\mathbf{z}^{(i+1)} = \arg \min_z \left( \frac{1}{2} \|\mathbf{x} - \mathbf{Dz}\|_2^2 + \frac{\rho}{2} \|\mathbf{z} - \mathbf{y}^{(i)} + \mathbf{u}^{(i)}\|_2^2 \right), \quad (61)$$

$$\mathbf{y}^{(i+1)} = \arg \min_y \left( \lambda \|\mathbf{y}\|_1 + \frac{\rho}{2} \|\mathbf{z}^{(i+1)} - \mathbf{y} + \mathbf{u}^{(i)}\|_2^2 \right), \quad (62)$$

$$\mathbf{u}^{(i+1)} = \mathbf{u}^{(i)} + \mathbf{z}^{(i+1)} - \mathbf{y}^{(i+1)}, \quad (63)$$

where  $\rho > 0$  is *penalty parameter* related to the convergent rate of the iterations.

## Solving minimization problem (6<sub>1</sub>)

---

Define

$$F_1(\mathbf{z}) := \frac{1}{2} \|\mathbf{x} - \mathbf{D}\mathbf{z}\|_2^2 + \frac{\rho}{2} \|\mathbf{z} - \mathbf{y}^{(i)} + \mathbf{u}^{(i)}\|_2^2.$$

Then  $F_1$  is a quadratic function in variables  $z_1, z_2, \dots, z_n$  and  $F_1(\mathbf{z}) \geq 0 \forall \mathbf{z} \in \mathbb{R}^n$ . To solve “ $\min_{\mathbf{z}} F_1(\mathbf{z})$ ”, first we compute

$$\begin{aligned} \nabla F_1(\mathbf{z}) &= -\mathbf{D}^\top (\mathbf{x} - \mathbf{D}\mathbf{z}) + \rho \mathbf{I} (\mathbf{z} - \mathbf{y}^{(i)} + \mathbf{u}^{(i)}) \\ &= (\mathbf{D}^\top \mathbf{D} + \rho \mathbf{I}) \mathbf{z} - (\mathbf{D}^\top \mathbf{x} + \rho (\mathbf{y}^{(i)} - \mathbf{u}^{(i)})). \end{aligned}$$

Letting  $\nabla F_1(\mathbf{z}) = \mathbf{0}$ , we have

$$(\mathbf{D}^\top \mathbf{D} + \rho \mathbf{I}) \mathbf{z} = (\mathbf{D}^\top \mathbf{x} + \rho (\mathbf{y}^{(i)} - \mathbf{u}^{(i)})).$$

Therefore, we obtain the solution

$$\mathbf{z}^{(i+1)} = (\mathbf{D}^\top \mathbf{D} + \rho \mathbf{I})^{-1} (\mathbf{D}^\top \mathbf{x} + \rho (\mathbf{y}^{(i)} - \mathbf{u}^{(i)})).$$

## Solving minimization problem (6<sub>2</sub>)

---

Using the *soft-thresholding function*  $\mathcal{S}_{\lambda/\rho}$ , the solution of problem (6<sub>2</sub>) has the closed form (see next few pages):

$$\mathbf{y}^{(i+1)} = \mathcal{S}_{\lambda/\rho}(\mathbf{z}^{(i+1)} + \mathbf{u}^{(i)}),$$

where *the soft-thresholding (軟閾值) function*  $\mathcal{S}_{\lambda/\rho}$ , is defined by

$$\mathcal{S}_{\lambda/\rho}(\mathbf{v}) = \text{sign}(\mathbf{v}) \odot \max(\mathbf{0}, |\mathbf{v}| - \lambda/\rho),$$

and  $\text{sign}(\cdot)$ ,  $\max(\cdot, \cdot)$ , and  $|\cdot|$  are all applied to the input vector  $\mathbf{v}$  component-wisely, and  $\odot$  is the Hadamard product.

Finally, the iterative scheme can be posed as follows:

$$\mathbf{z}^{(i+1)} = (\mathbf{D}^\top \mathbf{D} + \rho \mathbf{I})^{-1} (\mathbf{D}^\top \mathbf{x} + \rho(\mathbf{y}^{(i)} - \mathbf{u}^{(i)})), \quad (7_1)$$

$$\mathbf{y}^{(i+1)} = \mathcal{S}_{\lambda/\rho}(\mathbf{z}^{(i+1)} + \mathbf{u}^{(i)}), \quad (7_2)$$

$$\mathbf{u}^{(i+1)} = \mathbf{u}^{(i)} + \mathbf{z}^{(i+1)} - \mathbf{y}^{(i+1)}. \quad (7_3)$$

## Details of the solution of problem (6<sub>2</sub>)

---

Recall the problem (6<sub>2</sub>),

$$\mathbf{y}^{(i+1)} = \arg \min_{\mathbf{y}} \left( \lambda \|\mathbf{y}\|_1 + \frac{\rho}{2} \|\mathbf{z}^{(i+1)} - \mathbf{y} + \mathbf{u}^{(i)}\|_2^2 \right). \quad (6_2)$$

Let  $\mathbf{v} := \mathbf{z}^{(i+1)} + \mathbf{u}^{(i)} \in \mathbb{R}^n$ . Then we have

$$\mathbf{y}^{(i+1)} = \arg \min_{\mathbf{y}} \left( \lambda \|\mathbf{y}\|_1 + \frac{\rho}{2} \|\mathbf{v} - \mathbf{y}\|_2^2 \right).$$

Define a real-valued function  $F_2(\mathbf{y})$  as follows:

$$\begin{aligned} F_2(\mathbf{y}) &= \lambda \|\mathbf{y}\|_1 + \frac{\rho}{2} \|\mathbf{v} - \mathbf{y}\|_2^2 \\ &= \left( \lambda |y_1| + \frac{\rho}{2} (v_1 - y_1)^2 \right) + \cdots + \left( \lambda |y_n| + \frac{\rho}{2} (v_n - y_n)^2 \right) \\ &:= f_1(y_1) + \cdots + f_n(y_n), \end{aligned}$$

where we define

$$f_j(y) := \lambda |y| + \frac{\rho}{2} (v_j - y)^2 \quad \forall j = 1, 2, \dots, n.$$

## Analysis of functions $f_j$

---

For simplicity of the presentation, we consider the function

$$f(y) = \lambda|y| + \frac{\rho}{2}(v - y)^2.$$

Computing the derivative of  $f(y)$  for  $y \neq 0$ , we have

$$f'(y) = \begin{cases} \lambda - \rho(v - y) & \forall y > 0, \\ -\lambda - \rho(v - y) & \forall y < 0. \end{cases}$$

Let  $f'(y) = 0$ . Then we have

$$y = v - \frac{\lambda}{\rho} \text{ for } y > 0 \quad \text{and} \quad y = v + \frac{\lambda}{\rho} \text{ for } y < 0.$$

Therefore, the all critical numbers of  $f$  are given by

$$c = v - \frac{\lambda}{\rho} \text{ if } c > 0, \quad c = v + \frac{\lambda}{\rho} \text{ if } c < 0, \quad c = 0.$$

*In order to find the minimum of  $f$ , we consider the following three cases:*

$$v > \frac{\lambda}{\rho}, \quad v < -\frac{\lambda}{\rho}, \quad -\frac{\lambda}{\rho} \leq v \leq \frac{\lambda}{\rho}.$$

## Case 1: $v > \frac{\lambda}{\rho}$

---

In this case,  $c = v - \frac{\lambda}{\rho} > 0$  is the only critical number and

$$\begin{aligned} f(c) = f\left(v - \frac{\lambda}{\rho}\right) &= \lambda\left(v - \frac{\lambda}{\rho}\right) + \frac{\rho}{2}\left(v - \left(v - \frac{\lambda}{\rho}\right)\right)^2 \\ &= \frac{\rho}{2}\left(v^2 - \left(v - \frac{\lambda}{\rho}\right)^2\right) < \frac{\rho}{2}v^2 = f(0). \end{aligned}$$

For  $y \geq 0$ , since  $f$  is a quadratic polynomial in  $y$  with positive leading coefficient, we can conclude that  $f(c) \leq f(y)$  for all  $y \geq 0$ .

For  $y < 0$ ,  $f(y)$  is monotone decreasing since

$$\begin{aligned} f'(y) &= \lambda \operatorname{sign}(y) - \rho(v - y) = -\lambda - \rho v + \rho y \\ &< -\lambda - \lambda + \rho y = -2\lambda + \rho y < 0, \end{aligned}$$

which implies  $f(y) > f(0)$  for all  $y < 0$ .

*Therefore,  $f$  has a minimum at  $c = v - \frac{\lambda}{\rho} > 0$ .*

## Case 2: $v < -\frac{\lambda}{\rho}$

---

In this case,  $c = v + \frac{\lambda}{\rho} < 0$  is the only critical number and

$$\begin{aligned} f(c) = f\left(v + \frac{\lambda}{\rho}\right) &= -\lambda\left(v + \frac{\lambda}{\rho}\right) + \frac{\rho}{2}\left(v - \left(v + \frac{\lambda}{\rho}\right)\right)^2 \\ &= \frac{\rho}{2}\left(v^2 - \left(v + \frac{\lambda}{\rho}\right)^2\right) < \frac{\rho}{2}v^2 = f(0). \end{aligned}$$

For  $y \leq 0$ , since  $f$  is a quadratic polynomial in  $y$  with positive leading coefficient, we can conclude that  $f(c) \leq f(y)$  for all  $y \leq 0$ .

For  $y > 0$ ,  $f(y)$  is monotone increasing since

$$\begin{aligned} f'(y) &= \lambda \operatorname{sign}(y) - \rho(v - y) = \lambda - \rho v + \rho y \\ &> \lambda + \lambda + \rho y = 2\lambda + \rho y > 0, \end{aligned}$$

which implies  $f(y) > f(0)$  for all  $y > 0$ .

*Therefore,  $f$  has a minimum at  $c = v + \frac{\lambda}{\rho} > 0$ .*

### Case 3: $-\frac{\lambda}{\rho} \leq v \leq \frac{\lambda}{\rho}$

---

In this case, we have no critical number except the non-differentiable point  $y = 0$ .

For  $y > 0$ , we have

$$\begin{aligned} f'(y) &= \lambda - \rho(v - y) = \lambda - \rho v + \rho y \\ &\geq \lambda - \lambda + \rho y = \rho y > 0. \end{aligned}$$

Thus,  $f(y)$  is monotone increasing and then  $f(y) > f(0)$  for all  $y > 0$ .

For  $y < 0$ , we have

$$\begin{aligned} f'(y) &= -\lambda - \rho(v - y) = -\lambda - \rho v + \rho y \\ &\leq -\lambda + \lambda + \rho y = \rho y < 0. \end{aligned}$$

Thus,  $f(y)$  is monotone decreasing and then  $f(y) > f(0)$  for all  $y < 0$ .

*Therefore,  $f$  has a minimum at 0.*

## Solution of problem (6<sub>2</sub>)

---

By the above discussions, we have

$$\arg \min_y f(y) = \begin{cases} v + \frac{\lambda}{\rho}, & \text{if } v < -\frac{\lambda}{\rho}, & \text{(case 2)} \\ 0, & \text{if } |v| \leq \frac{\lambda}{\rho}, & \text{(case 3)} \\ v - \frac{\lambda}{\rho}, & \text{if } v > \frac{\lambda}{\rho}. & \text{(case 1)} \end{cases}$$

In other words, we have

$$\arg \min_y f(y) = \mathcal{S}_{\lambda/\rho}(v) = \text{sign}(v) \max(0, |v| - \lambda/\rho).$$

Therefore,

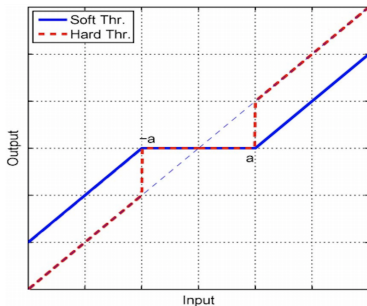
$$\mathbf{y}^{(i+1)} = \arg \min_y F_2(\mathbf{y}) = \mathcal{S}_{\lambda/\rho}(\mathbf{v}) = \mathcal{S}_{\lambda/\rho}(\mathbf{z}^{(i+1)} + \mathbf{u}^{(i)}).$$

where *the soft-thresholding function*,

$$\mathcal{S}_{\lambda/\rho}(\mathbf{v}) := \text{sign}(\mathbf{v}) \odot \max(\mathbf{0}, |\mathbf{v}| - \lambda/\rho),$$

and  $\text{sign}(\cdot)$ ,  $\max(\cdot, \cdot)$ , and  $|\cdot|$  are all applied to the input vector  $\mathbf{v}$  component-wisely, and  $\odot$  is the Hadamard product.

## Soft- and hard-thresholding functions



(This figure quoted from “M. Elad, M. A. T. Figueiredo, and Y. Ma, On the role of sparse and redundant representations in image processing, *Proceedings of the IEEE*, 98 (2010), pp. 972-982”)

- soft-thresholding function: continuous
- hard-thresholding function: discontinuous

## Application to signal denoising

- First, we construct a random dictionary matrix  $D \in \mathbb{R}^{512 \times 2048}$  and a random sparse vector  $z \in \mathbb{R}^{2048}$  with  $\|z\|_0 = 32$ . We then have the true signal  $x := Dz$ .
- Define the noise signal  $x_n := x + n$ , where  $n \in \mathbb{R}^{512}$  is a random white Gaussian noise with noised powers  $P = 0.5, 1, 5$ . We then consider  $\lambda = 5, 10, 20, 30$  for the minimization problem.
- *Peak signal-to-noise ratio (PSNR)*: We define the mean squared error (MSE) and then the PSNR as follows:

$$MSE := \frac{1}{512} \sum_{i=1}^{512} (\text{true}(i) - \text{approx}(i))^2,$$

$$PSNR := 10 \times \log_{10} \left( \frac{\max^2}{MSE} \right),$$

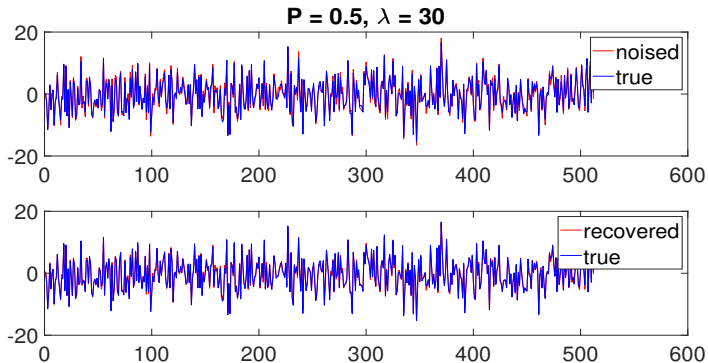
where “max” is the maximum amplitude of the true signal.

- Source of matlab code:

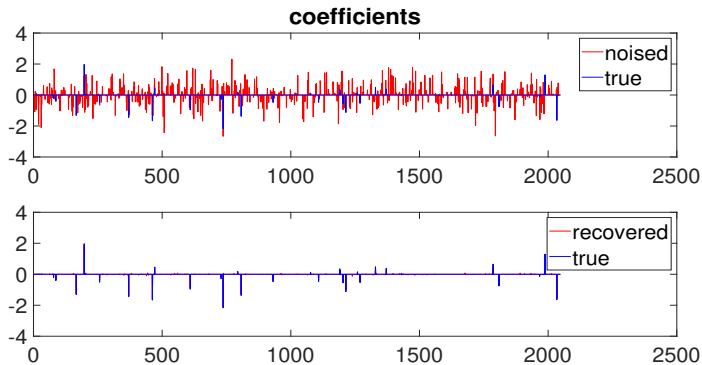
<http://brendt.wohlberg.net/software/SPORCO/>

## Numerical results for $P = 0.5$ and $\lambda = 30$

---

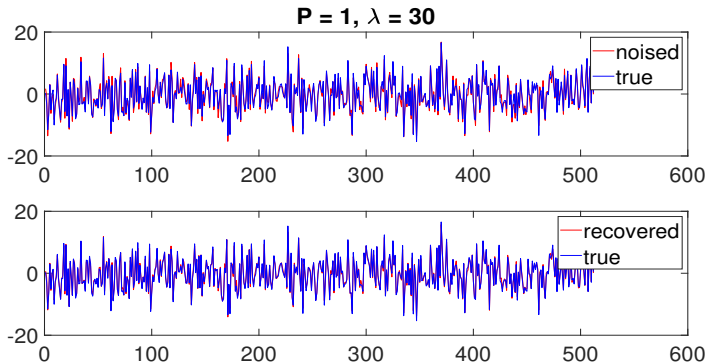


## Coefficients for $P = 0.5$ and $\lambda = 30$

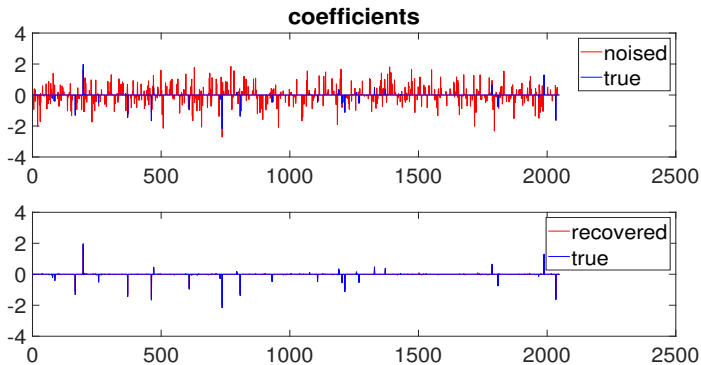


## Numerical results for $P = 1$ and $\lambda = 30$

---

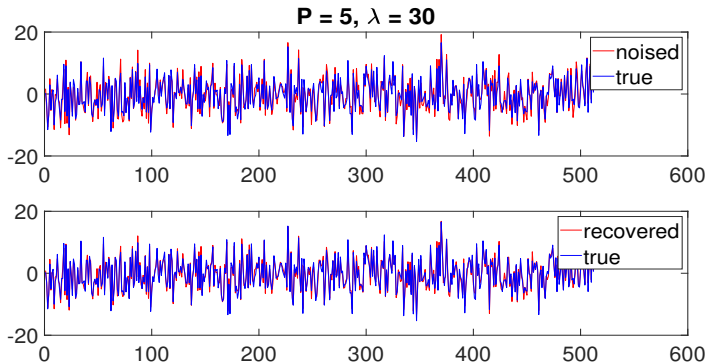


## Coefficients for $P = 1$ and $\lambda = 30$

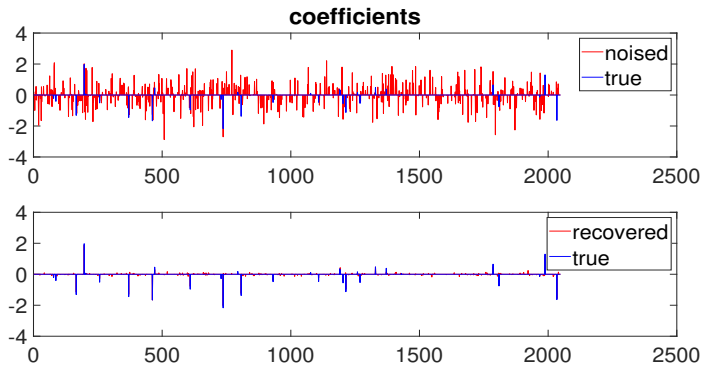


## Numerical result for $P = 5$ and $\lambda = 30$

---



## Coefficients for $P = 5$ and $\lambda = 30$



## PSNR values and iteration numbers

*In general, the higher the value of PSNR the better the quality of the recovered signals.*

PSNR values

$P$	0.5	0.5	1	1	5	5
$\lambda$	noised	rcvered	noised	rcvered	noised	rcvered
5	29.51	30.36	29.71	30.41	25.57	26.11
10	29.51	31.16	29.71	31.10	25.57	26.63
20	29.51	32.55	29.71	32.23	25.57	27.62
30	29.51	<b>33.45</b>	29.71	<b>32.77</b>	25.57	<b>28.50</b>

Iteration numbers of ADMM

$\lambda \backslash P$	0.5	1	5
5	550	664	569
10	301	303	320
20	172	169	186
30	<b>129</b>	<b>130</b>	<b>154</b>

## Sparse dictionary learning problem

---

*In the SR problem, the solution of interest  $\mathbf{z}^*$  is the coefficient vector of a linear combination of over-complete basis elements (columns) from a given dictionary  $\mathbf{D}$  under some sparsity constraint. Therefore, it is typically accompanied by a dictionary learning mechanism.*

We are going to study a more general problem. The dictionary  $\mathbf{D}$  is unknown and needed to be sought together with the sparse solution.

**SDL problem:** *Let  $\{\mathbf{x}_i\}_{i=1}^N \subset \mathbb{R}^m$  be a given dataset of signals. We seek a dictionary matrix  $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_n] \in \mathbb{R}^{m \times n}$  together with the sparse coefficient vectors  $\{\mathbf{z}_i\}_{i=1}^N \subset \mathbb{R}^n$  that solve the minimization problem:*

$$\min_{\mathbf{D}, \{\mathbf{z}_i\}} \left( \frac{1}{2} \sum_{i=1}^N \|\mathbf{x}_i - \mathbf{D}\mathbf{z}_i\|_2^2 + \lambda \sum_{i=1}^N \|\mathbf{z}_i\|_1 \right)$$

subject to  $\|\mathbf{d}_k\|_2 \leq 1, \forall 1 \leq k \leq n,$

*where  $\lambda > 0$  is a penalty parameter.*

## Problem formulation in a more compact form

---

To simplify the formulation of the SDL problem, we define

$$\begin{aligned} \mathbf{X} &= [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N] \in \mathbb{R}^{m \times N}, \\ \mathbf{Z} &= [\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_N] \in \mathbb{R}^{n \times N}. \end{aligned}$$

Then the SDL problem can be posed as follows: *Given a training data matrix  $\mathbf{X}$ , find a dictionary matrix  $\mathbf{D}$  and a coefficient matrix  $\mathbf{Z}$  such that*

$$\begin{aligned} \min_{\mathbf{D}, \mathbf{Z}} \left( \frac{1}{2} \|\mathbf{X} - \mathbf{D}\mathbf{Z}\|_F^2 + \lambda \|\mathbf{Z}\|_{1,1} \right) \quad (\star\star) \\ \text{subject to } \|\mathbf{d}_k\|_2 \leq 1, \quad \forall 1 \leq k \leq n, \end{aligned}$$

where  $\|\cdot\|_F$  denotes the Frobenius norm and  $\|\mathbf{Z}\|_{1,1}$  is the  $L_{1,1}$ -norm which is defined as

$$\|\mathbf{Z}\|_{1,1} := \sum_{i=1}^N \|\mathbf{z}_i\|_1.$$

## An iterative approach for solving the SDL problem

---

In the SDL problem ( $\star\star$ ), we have two unknown matrices  $D$  and  $Z$ . We will use a simple iterative approach together with the ADMM to solve ( $\star\star$ ), though it is more complicated.

Given an initial guess  $D_{(0)}$ , for  $j = 0, 1, \dots$ , we solve the following two sub-problems alternately:

$$Z_{(j)} = \arg \min_Z \left( \frac{1}{2} \|X - D_{(j)} Z\|_F^2 + \lambda \|Z\|_{1,1} \right), \quad (8)$$

$$D_{(j+1)} = \arg \min_D \left( \frac{1}{2} \|X - D Z_{(j)}\|_F^2 + \lambda \|Z_{(j)}\|_{1,1} \right) \\ \text{subject to } \|d_k\|_2 \leq 1, \forall 1 \leq k \leq n. \quad (9)$$

We iterate (8) and (9) until convergence is achieved. As we have introduced previously, problems (8) and (9) will be solved by ADMM.

## ADMM for solving problem (8)

---

- Adding an *auxiliary variable*  $\mathbf{Y}$  and a *dual variable*  $\mathbf{U}$ , we define

$$f(\mathbf{Z}) := \frac{1}{2} \|\mathbf{X} - \mathbf{D}_{(j)}\mathbf{Z}\|_F^2, \quad g(\mathbf{Y}) := \lambda \|\mathbf{Y}\|_{1,1}, \quad \mathbf{Z} = \mathbf{Y}.$$

- Then the ADMM for solving (8) is given by

$$\mathbf{Z}^{(i+1)} = \arg \min_{\mathbf{Z}} \left( \frac{1}{2} \|\mathbf{X} - \mathbf{D}_{(j)}\mathbf{Z}\|_F^2 + \frac{\rho}{2} \|\mathbf{Z} - \mathbf{Y}^{(i)} + \mathbf{U}^{(i)}\|_F^2 \right), \quad (8_1)$$

$$\mathbf{Y}^{(i+1)} = \arg \min_{\mathbf{Y}} \left( \lambda \|\mathbf{Y}\|_{1,1} + \frac{\rho}{2} \|\mathbf{Z}^{(i+1)} - \mathbf{Y} + \mathbf{U}^{(i)}\|_F^2 \right), \quad (8_2)$$

$$\mathbf{U}^{(i+1)} = \mathbf{U}^{(i)} + \mathbf{Z}^{(i+1)} - \mathbf{Y}^{(i+1)}. \quad (8_3).$$

- Similar to the SR problem, we will use the same methods to solve the sub-problems (8<sub>1</sub>) and (8<sub>2</sub>).

## Solving minimization problem (8<sub>1</sub>)

---

Define

$$F_1(\mathbf{Z}) := \frac{1}{2} \|\mathbf{X} - \mathbf{D}_{(j)}\mathbf{Z}\|_F^2 + \frac{\rho}{2} \|\mathbf{Z} - \mathbf{Y}^{(i)} + \mathbf{U}^{(i)}\|_F^2.$$

To solve “ $\min_{\mathbf{Z}} F_1(\mathbf{Z})$ ”, first we compute

$$\begin{aligned} \nabla F_1(\mathbf{Z}) &= -\mathbf{D}_{(j)}^\top (\mathbf{X} - \mathbf{D}_{(j)}\mathbf{Z}) + \rho \mathbf{I} (\mathbf{Z} - \mathbf{Y}^{(i)} + \mathbf{U}^{(i)}) \\ &= (\mathbf{D}_{(j)}^\top \mathbf{D}_{(j)} + \rho \mathbf{I})\mathbf{Z} - (\mathbf{D}_{(j)}^\top \mathbf{X} + \rho(\mathbf{Y}^{(i)} - \mathbf{U}^{(i)})). \end{aligned}$$

Letting  $\nabla F_1(\mathbf{Z}) = 0$ , we have

$$(\mathbf{D}_{(j)}^\top \mathbf{D}_{(j)} + \rho \mathbf{I})\mathbf{Z} = (\mathbf{D}_{(j)}^\top \mathbf{X} + \rho(\mathbf{Y}^{(i)} - \mathbf{U}^{(i)})).$$

Therefore, we obtain the solution

$$\mathbf{Z}^{(i+1)} = (\mathbf{D}_{(j)}^\top \mathbf{D}_{(j)} + \rho \mathbf{I})^{-1} (\mathbf{D}_{(j)}^\top \mathbf{X} + \rho(\mathbf{Y}^{(i)} - \mathbf{U}^{(i)})).$$

## Solving minimization problem (8<sub>2</sub>)

---

Using the component-wise soft-thresholding function, the solution of problem (8<sub>2</sub>) has the closed form:

$$\mathbf{Y}^{(i+1)} = \mathcal{S}_{\lambda/\rho}(\mathbf{Z}^{(i+1)} + \mathbf{U}^{(i)}),$$

where

$$\mathcal{S}_{\lambda/\rho}(\mathbf{V}) = \text{sign}(\mathbf{V}) \odot \max(\mathbf{0}, |\mathbf{V}| - \lambda/\rho),$$

with  $\text{sign}(\mathbf{V})$  and  $|\mathbf{V}|$  are element-wisely applied to the matrix  $\mathbf{V}$  and  $\odot$  is the Hadamard product.

Therefore, the iterative scheme can be posed as follows:

$$\mathbf{Z}^{(i+1)} = (\mathbf{D}_{(j)}^\top \mathbf{D}_{(j)} + \rho \mathbf{I})^{-1} (\mathbf{D}_{(j)}^\top \mathbf{X} + \rho (\mathbf{Y}^{(i)} - \mathbf{U}^{(i)})), \quad (10_1)$$

$$\mathbf{Y}^{(i+1)} = \mathcal{S}_{\lambda/\rho}(\mathbf{Z}^{(i+1)} + \mathbf{U}^{(i)}), \quad (10_2)$$

$$\mathbf{U}^{(i+1)} = \mathbf{U}^{(i)} + \mathbf{Z}^{(i+1)} - \mathbf{Y}^{(i+1)}. \quad (10_3)$$

## Solving minimization problem (9)

---

Recall that

$$\begin{aligned} \mathbf{D}_{(j+1)} &= \arg \min_{\mathbf{D}} \left( \frac{1}{2} \|\mathbf{X} - \mathbf{D}\mathbf{Z}_{(j)}\|_F^2 + \lambda \|\mathbf{Z}_{(j)}\|_{1,1} \right) \\ &\text{subject to } \|\mathbf{d}_k\|_2 \leq 1, \forall 1 \leq k \leq n. \end{aligned} \quad (9)$$

Since the term  $\lambda \|\mathbf{Z}_{(j)}\|_{1,1}$  is a fixed number when  $\mathbf{Z}_{(j)}$  is given, problem (9) can be replaced by

$$\begin{aligned} \mathbf{D}_{(j+1)} &= \arg \min_{\mathbf{D}} \frac{1}{2} \|\mathbf{X} - \mathbf{D}\mathbf{Z}_{(j)}\|_F^2 \\ &\text{subject to } \|\mathbf{d}_k\|_2 \leq 1, \forall 1 \leq k \leq n. \end{aligned} \quad (9')$$

Next, we introduce an *auxiliary variable*  $\mathbf{G}$  and a *dual variable*  $\mathbf{H}$  in ADMM for solving (9').

## ADMM for solving problem (9')

---

Define

$$\begin{aligned}g(\mathbf{G}) &:= \{[\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_n] : \|\mathbf{d}_k\|_2 \leq 1, \forall 1 \leq k \leq n\}, \\ \mathbf{G} &:= \mathbf{D}.\end{aligned}$$

The ADMM for solving problem (9') is given by

$$\mathbf{D}^{(i+1)} = \arg \min_{\mathbf{D}} \left( \frac{1}{2} \|\mathbf{X} - \mathbf{D}\mathbf{Z}_{(j)}\|_F^2 + \frac{\rho}{2} \|\mathbf{D} - \mathbf{G}^{(i)} + \mathbf{H}^{(i)}\|_F^2 \right), \quad (9_1)$$

$$\mathbf{G}^{(i+1)} = \text{proj}_{g(\mathbf{G})} \{\mathbf{D}^{(i+1)}\}, \quad (9_2)$$

$$\mathbf{H}^{(i+1)} = \mathbf{H}^{(i)} + \mathbf{D}^{(i+1)} - \mathbf{G}^{(i+1)}. \quad (9_3)$$

For solving problem (9<sub>1</sub>), we define

$$F_2(\mathbf{D}) := \frac{1}{2} \|\mathbf{X} - \mathbf{D}\mathbf{Z}_{(j)}\|_F^2 + \frac{\rho}{2} \|\mathbf{D} - \mathbf{G}^{(i)} + \mathbf{H}^{(i)}\|_F^2.$$

## Solving minimization problem (9<sub>1</sub>)

---

Computing  $\nabla F_2(D)$ , we have

$$\begin{aligned}\nabla F_2(D) &= (\mathbf{X} - \mathbf{D}\mathbf{Z}_{(j)})(-\mathbf{Z}_{(j)}^\top) + \rho\mathbf{I}_m(\mathbf{D} - \mathbf{G}^{(i)} + \mathbf{H}^{(i)}) \\ &= \mathbf{D}(\rho\mathbf{I}_n + \mathbf{Z}_{(j)}\mathbf{Z}_{(j)}^\top) + \mathbf{X}\mathbf{Z}_{(j)}^\top - \rho(\mathbf{G}^{(i)} - \mathbf{H}^{(i)}).\end{aligned}$$

Letting  $\nabla F_2(D) = \mathbf{0}$ , we have

$$\mathbf{D}(\mathbf{Z}_{(j)}\mathbf{Z}_{(j)}^\top + \rho\mathbf{I}_n) = \mathbf{X}\mathbf{Z}_{(j)}^\top - \rho(\mathbf{G}^{(i)} - \mathbf{H}^{(i)}).$$

Therefore, we obtain the solution

$$\mathbf{D}^{(i+1)} = (\mathbf{X}\mathbf{Z}_{(j)}^\top - \rho(\mathbf{G}^{(i)} - \mathbf{H}^{(i)}))(\mathbf{Z}_{(j)}\mathbf{Z}_{(j)}^\top + \rho\mathbf{I}_n)^{-1}.$$

Finally, the ADMM for problem (9') is given by

$$\mathbf{D}^{(i+1)} = (\mathbf{X}\mathbf{Z}_{(j)}^\top - \rho(\mathbf{G}^{(i)} - \mathbf{H}^{(i)}))(\mathbf{Z}_{(j)}\mathbf{Z}_{(j)}^\top + \rho\mathbf{I}_n)^{-1}, \quad (11_1)$$

$$\mathbf{G}^{(i+1)} = \text{proj}_{\mathcal{G}}\{\mathbf{D}^{(i+1)}\}, \quad (11_2)$$

$$\mathbf{H}^{(i+1)} = \mathbf{H}^{(i)} + \mathbf{D}^{(i+1)} - \mathbf{G}^{(i+1)}. \quad (11_3)$$

## Convergence and stopping criterion

---

- In [Boyd *et al.* 2010], there are more details about convergence results of the ADMM.
- In the iterative scheme (10<sub>1</sub>), (10<sub>2</sub>), (10<sub>3</sub>), we define

$$\mathbf{R}_z = \mathbf{Y}^{(i+1)} - \mathbf{Y}^{(i)}, \quad \mathbf{S}_z = \mathbf{U}^{(i+1)} - \mathbf{U}^{(i)}.$$

If  $\mathbf{R}_z$  and  $\mathbf{S}_z$  less than the tolerances  $\varepsilon_{R_z}$  and  $\varepsilon_{S_z}$ , then we say that the iteration of coefficients  $\mathbf{Z}^{(i+1)}$  converges.

In the iterative scheme (11<sub>1</sub>), (11<sub>2</sub>), (11<sub>3</sub>), we define

$$\mathbf{R}_d = \mathbf{G}^{(i+1)} - \mathbf{G}^{(i)}, \quad \mathbf{S}_d = \mathbf{H}^{(i+1)} - \mathbf{H}^{(i)}.$$

If  $\mathbf{R}_d$  and  $\mathbf{S}_d$  less than the tolerances  $\varepsilon_{R_d}$  and  $\varepsilon_{S_d}$ , then we say that the iteration of dictionary  $\mathbf{D}^{(i+1)}$  converges.

## References and source codes

---

### $\ell^1$ - $\ell^0$ equivalence problem:

- [1] D. L. Donoho, For most large underdetermined systems of linear equations the minimal  $\ell_1$ -norm solution is also the sparsest solution, *Communications on Pure and Applied Mathematics*, 59 (2006), pp. 797-829.
- [2] Y. Sharon, J. Wright, and Y. Ma, Computation and relaxation of conditions for equivalence between  $\ell^1$  and  $\ell^0$  minimization, *UIUC Technical Report UILU-ENG-07-2008*, 2007.

### ADMM:

- [1] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, Distributed optimization and statistical learning via the ADMM, *Foundations and Trends<sup>®</sup> in Machine Learning*, 3 (2010), pp. 1-122.

### Sparse dictionary learning:

[https://en.wikipedia.org/wiki/Sparse\\_dictionary\\_learning](https://en.wikipedia.org/wiki/Sparse_dictionary_learning)

**Matlab codes:** <http://brendt.wohlberg.net/software/SPORCO/>

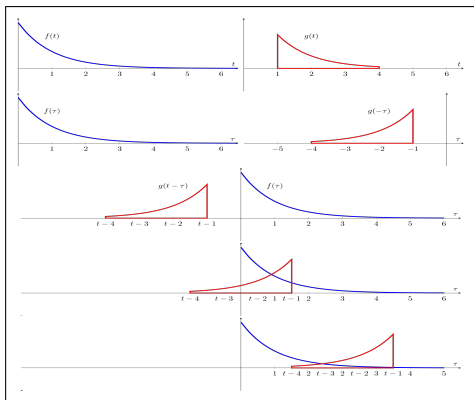
## Part II

# Convolutional Sparse Representation and Dictionary Learning

## Convolution of two functions

Let  $f$  and  $g$  be two integrable functions with compact supports in  $\mathbb{R}$ . Then the convolution of  $f$  and  $g$  is defined as a function in variable  $t$ ,

$$(f * g)(t) := \int_{-\infty}^{\infty} f(\tau)g(t - \tau) d\tau, \quad t \in \mathbb{R}.$$



## Convolution of two vectors

**Definition:** Let  $\mathbf{u} = [u_1, \dots, u_n]^\top \in \mathbb{R}^n$  and  $\mathbf{v} = [v_1, \dots, v_m]^\top \in \mathbb{R}^m$ . The convolution of  $\mathbf{u}$  and  $\mathbf{v}$ , denoted by  $\mathbf{u} * \mathbf{v}$ , is defined as follows:

$$\mathbf{u} * \mathbf{v} := \begin{bmatrix} u_1 v_1 \\ u_1 v_2 + u_2 v_1 \\ u_1 v_3 + u_2 v_2 + u_3 v_1 \\ \vdots \\ u_{n-2} v_m + u_{n-1} v_{m-1} + u_n v_{m-2} \\ u_{n-1} v_m + u_n v_{m-1} \\ u_n v_m \end{bmatrix} \in \mathbb{R}^{m+n-1}.$$

More specifically, for  $i = 1, 2, \dots, (m + n - 1)$ , the  $i$ -th component of  $\mathbf{u} * \mathbf{v}$  is given by

$$(\mathbf{u} * \mathbf{v})_i = \sum_{j=\max(1, i-m+1)}^{\min(i, n)} u_j v_{i-j+1}.$$

**Remark:** Convolutional operator  $*$  is commutative, i.e.,  $\mathbf{u} * \mathbf{v} = \mathbf{v} * \mathbf{u}$ .

## Convolutional sparse representation (CSR) problem

**CSR problem:** Given a signal  $x \in \mathbb{R}^m$  and a dictionary  $D = [d_1, \dots, d_n] \in \mathbb{R}^{\ell \times n}$ , we seek a sparse matrix  $Z = [z_1, \dots, z_n] \in \mathbb{R}^{k \times n}$ ,  $m = \ell + k - 1$ , which solves the following minimization problem:

$$\min_Z \left( \frac{1}{2} \left\| x - \sum_{j=1}^n d_j * z_j \right\|_2^2 + \lambda \sum_{j=1}^n \|z_j\|_1 \right),$$

where  $\lambda > 0$  is a penalty parameter.

### Remarks:

- In SR, we use  $Dz$  to recover the signal  $x$ ,

$$x \approx Dz = d_1 z_1 + d_2 z_2 + \dots + d_n z_n = \sum_{j=1}^n d_j z_j.$$

In CSR, we use  $\sum_{j=1}^n d_j * z_j$  instead,

$$x \approx d_1 * z_1 + d_2 * z_2 + \dots + d_n * z_n = \sum_{j=1}^n d_j * z_j.$$

- Convolution is a way to regulate  $d_j * z_j$  such that  $x \approx \sum_{j=1}^n d_j * z_j$ . It is more flexible than  $x \approx \sum_{j=1}^n d_j z_j$ , but indeed more expensive!

## Toeplitz matrix

We define an  $(m + n - 1) \times m$  matrix  $\mathbf{U}$  in terms of  $u_i$ , which is called a *Toeplitz matrix*, as follows:

$$\mathbf{U} := \begin{bmatrix} u_1 & 0 & \cdots & 0 & 0 \\ u_2 & u_1 & \ddots & 0 & 0 \\ \vdots & u_2 & \ddots & 0 & 0 \\ \vdots & \vdots & \ddots & u_1 & 0 \\ u_{n-1} & \vdots & \vdots & u_2 & u_1 \\ u_n & u_{n-1} & \vdots & \vdots & u_2 \\ 0 & u_n & \ddots & \vdots & \vdots \\ 0 & 0 & \ddots & u_{n-1} & \vdots \\ \vdots & \vdots & \ddots & u_n & u_{n-1} \\ 0 & 0 & \cdots & 0 & u_n \end{bmatrix}_{(m+n-1) \times m}$$

Then one can check that  $\mathbf{u} * \mathbf{v} = \mathbf{U}\mathbf{v}$ , where  $\mathbf{u} = [u_1, \dots, u_n]^\top \in \mathbb{R}^n$  and  $\mathbf{v} = [v_1, \dots, v_m]^\top \in \mathbb{R}^m$ .

## CSR problem using Toeplitz matrices

---

With the help of Toeplitz matrix, we can rewrite the CSR problem as

$$\min_{\tilde{\mathbf{z}}} \left( \frac{1}{2} \|\mathbf{x} - \tilde{\mathbf{D}}\tilde{\mathbf{z}}\|_2^2 + \lambda \|\tilde{\mathbf{z}}\|_1 \right), \quad (12)$$

with

$$\tilde{\mathbf{z}} = [\mathbf{z}_1^\top, \mathbf{z}_2^\top, \dots, \mathbf{z}_n^\top]^\top_{nk \times 1} \quad \text{and} \quad \tilde{\mathbf{D}} = [\mathbf{D}_1, \mathbf{D}_2, \dots, \mathbf{D}_n]_{(\ell+k-1) \times nk},$$

where  $\mathbf{D}_j$  is a Toeplitz  $(\ell + k - 1) \times k$  matrix associated with the column vector  $\mathbf{d}_j \in \mathbb{R}^\ell$ , and  $\ell + k - 1 = m$ .

### Remarks:

- We can use the same way for SR problem to solve the CSR problem (12). We can employ the ADMM, but it is too expensive since the matrix size of  $\tilde{\mathbf{D}}$  is too large.
- *The discrete Fourier transform  $\mathcal{F} : \mathbb{C}^N \rightarrow \mathbb{C}^N$  can help us to address this computational issue.*

## Discrete Fourier transform (DFT) and its inverse (IDFT)

- $\hat{\mathbf{x}} = \mathcal{F}(\mathbf{x})$ : The DFT  $\mathcal{F} : \mathbb{C}^N \rightarrow \mathbb{C}^N$  transforms a finite vector  $\mathbf{x} = [x_1, x_2, \dots, x_N]^\top$  into another vector  $\hat{\mathbf{x}} = [\hat{x}_1, \hat{x}_2, \dots, \hat{x}_N]^\top$ , which is defined by

$$\hat{x}_k = \sum_{n=1}^N x_n e^{-\frac{i2\pi}{N}(k-1)(n-1)}.$$

*Then DFT is an invertible linear transformation.*

- $\mathbf{x} = \mathcal{F}^{-1}(\hat{\mathbf{x}})$ : The inverse discrete Fourier transform (IDFT)  $\mathcal{F}^{-1} : \mathbb{C}^N \rightarrow \mathbb{C}^N, \hat{\mathbf{x}} \mapsto \mathbf{x}$ , is given by

$$x_n = \frac{1}{N} \sum_{k=1}^N \hat{x}_k e^{\frac{i2\pi}{N}(k-1)(n-1)}.$$

- Euler's formula:  $e^{i\theta} = \cos \theta + i \sin \theta, \forall \theta \in \mathbb{R}$ .

[https://en.wikipedia.org/wiki/Discrete\\_Fourier\\_transform](https://en.wikipedia.org/wiki/Discrete_Fourier_transform)

## Hadamard product

- Let  $\mathbf{u} \in \mathbb{R}^n$  and  $\mathbf{v} \in \mathbb{R}^m$ . Then  $\mathbf{u} * \mathbf{v} \in \mathbb{R}^{m+n-1}$  and

$$\mathcal{F}(\mathbf{u} * \mathbf{v}) = \mathcal{F}(\mathbf{u}') \odot \mathcal{F}(\mathbf{v}'),$$

where  $\mathcal{F}$  denotes the DFT,  $\mathbf{u}'$  and  $\mathbf{v}'$  are respectively *the zero padding of  $\mathbf{u}$  and  $\mathbf{v}$  with the same size of  $\mathbf{u} * \mathbf{v}$ , i.e.,*

$$\mathbf{u}' = [\mathbf{u}^\top, 0, \dots, 0]^\top, \quad \mathbf{v}' = [\mathbf{v}^\top, 0, \dots, 0]^\top \in \mathbb{R}^{m+n-1},$$

and  $\odot$  is the Hadamard product.

- The Hadamard product  $\odot$  of two vectors is a component-wise product. Let  $\mathbf{u} = [u_1, u_2, \dots, u_n]^\top$ ,  $\mathbf{v} = [v_1, v_2, \dots, v_n]^\top \in \mathbb{R}^n$ ,

$$\mathbf{u} \odot \mathbf{v} := [u_1 v_1, u_2 v_2, \dots, u_n v_n]^\top.$$

We can define a diagonal matrix  $\mathbf{U}$  such that  $\mathbf{u} \odot \mathbf{v} = \mathbf{U}\mathbf{v}$ , where

$$\mathbf{U} := \begin{bmatrix} u_1 & 0 & \cdots & 0 \\ 0 & u_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & u_n \end{bmatrix}.$$

## Recalling the CSR problem

**CSR problem:** Given  $\mathbf{x} \in \mathbb{R}^m$  and  $\mathbf{D} = [\mathbf{d}_1, \dots, \mathbf{d}_n] \in \mathbb{R}^{\ell \times n}$ , we seek  $\mathbf{Z} = [\mathbf{z}_1, \dots, \mathbf{z}_n] \in \mathbb{R}^{k \times n}$  with  $m = \ell + k - 1$  solving

$$\min_{\mathbf{Z}} \left( \frac{1}{2} \|\mathbf{x} - \sum_{j=1}^n \mathbf{d}_j * \mathbf{z}_j\|_2^2 + \lambda \sum_{j=1}^n \|\mathbf{z}_j\|_1 \right).$$

To solve the above minimization problem, we first use the ADMM algorithm to split it into three subproblems:

$$\mathbf{Z}^{(i+1)} = \arg \min_{\mathbf{Z}} \left( \frac{1}{2} \|\mathbf{x} - \sum_{j=1}^n \mathbf{d}_j * \mathbf{z}_j\|_2^2 + \frac{\rho}{2} \sum_{j=1}^n \|\mathbf{z}_j - \mathbf{y}_j^{(i)} + \mathbf{u}_j^{(i)}\|_2^2 \right),$$

$$\mathbf{Y}^{(i+1)} = \arg \min_{\mathbf{Y}} \left( \lambda \sum_{j=1}^n \|\mathbf{y}_j\|_1 + \frac{\rho}{2} \sum_{j=1}^n \|\mathbf{z}_j^{(i+1)} - \mathbf{y}_j + \mathbf{u}_j^{(i)}\|_2^2 \right)$$

$$\mathbf{U}^{(i+1)} = \mathbf{U}^{(i)} + \mathbf{Z}^{(i+1)} - \mathbf{Y}^{(i+1)}.$$

## Using discrete Fourier transform for $\mathbf{Z}$

We will use the discrete Fourier transform and Hadamard product to solve the subproblem of  $\mathbf{Z}$ . We can rewrite these subproblems as

$$\widehat{\mathbf{Z}}^{(i+1)} = \arg \min_{\widehat{\mathbf{Z}}} \left( \frac{1}{2} \|\widehat{\mathbf{x}} - \sum_{j=1}^n \widehat{\mathbf{d}}'_j \odot \widehat{\mathbf{z}}'_j\|_2^2 + \frac{\rho}{2} \sum_{j=1}^n \|\widehat{\mathbf{z}}'_j - \widehat{\mathbf{y}}'_j^{(i)} + \widehat{\mathbf{u}}'_j^{(i)}\|_2^2 \right),$$

$$\mathbf{Y}^{(i+1)} = \arg \min_{\mathbf{Y}} \left( \lambda \sum_{j=1}^n \|\mathbf{y}_j\|_1 + \frac{\rho}{2} \sum_{j=1}^n \|\mathcal{F}^{-1}(\widehat{\mathbf{z}}'_j^{(i+1)}) - \mathbf{y}_j + \mathbf{u}_j^{(i)}\|_2^2 \right),$$

$$\mathbf{U}^{(i+1)} = \mathbf{U}^{(i)} + \mathcal{F}^{-1}(\widehat{\mathbf{Z}}^{(i+1)}) - \mathbf{Y}^{(i+1)},$$

where

$$\mathcal{F}^{-1}(\widehat{\mathbf{Z}}) = [\mathcal{F}^{-1}(\widehat{\mathbf{z}}'_1), \mathcal{F}^{-1}(\widehat{\mathbf{z}}'_2), \dots, \mathcal{F}^{-1}(\widehat{\mathbf{z}}'_n)].$$

## Why can we use the discrete Fourier transform?

Note that the discrete Fourier transform  $\mathcal{F}$  is linear. Thus, we have

$$\begin{aligned} \frac{1}{2} \left\| \mathbf{x} - \sum_{j=1}^n \mathbf{d}_j * \mathbf{z}_j \right\|_2^2 &= \frac{1}{2m} \left\| \mathcal{F} \left( \mathbf{x} - \sum_{j=1}^n \mathbf{d}_j * \mathbf{z}_j \right) \right\|_2^2 \quad (\text{Plancherel theorem}) \\ &= \frac{1}{2m} \left\| \mathcal{F}(\mathbf{x}) - \mathcal{F} \left( \sum_{j=1}^n \mathbf{d}_j * \mathbf{z}_j \right) \right\|_2^2 \\ &= \frac{1}{2m} \left\| \widehat{\mathbf{x}} - \sum_{j=1}^n \mathcal{F}(\mathbf{d}_j * \mathbf{z}_j) \right\|_2^2 = \frac{1}{2m} \left\| \widehat{\mathbf{x}} - \sum_{j=1}^n \widehat{\mathbf{d}}_j \odot \widehat{\mathbf{z}}_j \right\|_2^2. \end{aligned}$$

Similarly, the second term of subproblem  $\mathbf{Z}$  can be rewritten as

$$\begin{aligned} \frac{\rho}{2} \sum_{j=1}^n \left\| \mathbf{z}_j - \mathbf{y}_j^{(i)} + \mathbf{u}_j^{(i)} \right\|_2^2 &= \frac{\rho}{2} \sum_{j=1}^n \left\| \mathbf{z}'_j - \mathbf{y}'_j{}^{(i)} + \mathbf{u}'_j{}^{(i)} \right\|_2^2 \\ &= \frac{\rho}{2m} \sum_{j=1}^n \left\| \widehat{\mathbf{z}}_j - \widehat{\mathbf{y}}_j{}^{(i)} + \widehat{\mathbf{u}}_j{}^{(i)} \right\|_2^2. \end{aligned}$$

**Note:**  $\mathbf{x} \in \mathbb{R}^m$ ,  $\mathbf{d}_j \in \mathbb{R}^\ell$ ,  $\mathbf{z}_j \in \mathbb{R}^k$ ,  $\mathbf{d}_j * \mathbf{z}_j \in \mathbb{R}^{\ell+k-1} = \mathbb{R}^m$ ,  $\mathbf{d}'_j, \mathbf{z}'_j \in \mathbb{R}^m$ .

## The subproblem of $\hat{Z}$

We first define

$$\hat{D}_j = \text{diag}(\hat{d}'_j). \quad (m \times m \text{ diagonal matrix})$$

Then the subproblem in the Fourier domain can be posed as:

$$\hat{z}^{(i+1)} = \arg \min_{\hat{z}} \left( \frac{1}{2} \|\hat{x} - \hat{D}\hat{z}\|_2^2 + \frac{\rho}{2} \|\hat{z} - \hat{y}^{(i)} + \hat{u}^{(i)}\|_2^2 \right),$$

where

$$\hat{D} = [\hat{D}_1, \hat{D}_2, \dots, \hat{D}_n]_{m \times mn}, \quad \hat{z} = [\hat{z}'_1{}^\top, \hat{z}'_2{}^\top, \dots, \hat{z}'_n{}^\top]^\top_{mn \times 1},$$

and

$$\hat{y} = [\hat{y}'_1{}^\top, \hat{y}'_2{}^\top, \dots, \hat{y}'_n{}^\top]^\top_{mn \times 1}, \quad \hat{u} = [\hat{u}'_1{}^\top, \hat{u}'_2{}^\top, \dots, \hat{u}'_n{}^\top]^\top_{mn \times 1}.$$

*Note that we drop the scalar factor  $1/m$  in the subproblem.*

## Rewriting the subproblems of ADMM

---

Using the above definitions, we can rewrite the subproblems of ADMM as:

$$\hat{\mathbf{z}}^{(i+1)} = \min_{\hat{\mathbf{z}}} \left( \frac{1}{2} \|\hat{\mathbf{x}} - \hat{\mathbf{D}}\hat{\mathbf{z}}\|_2^2 + \frac{\rho}{2} \|\hat{\mathbf{z}} - \hat{\mathbf{y}}^{(i)} + \hat{\mathbf{u}}^{(i)}\|_2^2 \right), \quad (13_1)$$

$$\mathbf{y}^{(i+1)} = \min_{\mathbf{y}} \left( \lambda \|\mathbf{y}\|_1 + \frac{\rho}{2} \sum_{j=1}^n \|\mathcal{F}^{-1}(\hat{\mathbf{z}}^{(i+1)}) - \mathbf{y} + \mathbf{u}^{(i)}\|_2^2 \right), \quad (13_2)$$

$$\mathbf{u}^{(i+1)} = \mathbf{u}^{(i)} + \mathcal{F}^{-1}(\hat{\mathbf{z}}^{(i+1)}) - \mathbf{y}^{(i+1)}, \quad (13_3)$$

where

$$\mathbf{y} = [\mathbf{y}'_1{}^\top, \mathbf{y}'_2{}^\top, \dots, \mathbf{y}'_n{}^\top]^\top_{mn \times 1}, \quad \mathbf{u} = [\mathbf{u}'_1{}^\top, \mathbf{u}'_2{}^\top, \dots, \mathbf{u}'_n{}^\top]^\top_{mn \times 1}.$$

## Solving minimization problem (13<sub>1</sub>)

---

First we define

$$F(\hat{\mathbf{z}}) = \frac{1}{2} \|\hat{\mathbf{x}} - \hat{\mathbf{D}}\hat{\mathbf{z}}\|_2^2 + \frac{\rho}{2} \|\hat{\mathbf{z}} - \hat{\mathbf{y}}^{(i)} + \hat{\mathbf{u}}^{(i)}\|_2^2.$$

To solve  $\min_{\hat{\mathbf{z}}} F(\hat{\mathbf{z}})$ , we compute

$$\begin{aligned} \nabla F(\hat{\mathbf{z}}) &= -\hat{\mathbf{D}}^\top (\hat{\mathbf{x}} - \hat{\mathbf{D}}\hat{\mathbf{z}}) + \rho \mathbf{I} (\hat{\mathbf{z}} - \hat{\mathbf{y}}^{(i)} + \hat{\mathbf{u}}^{(i)}) \\ &= (\hat{\mathbf{D}}^\top \hat{\mathbf{D}} + \rho \mathbf{I}) \hat{\mathbf{z}} - (\hat{\mathbf{D}}^\top \hat{\mathbf{x}} + \rho(\hat{\mathbf{y}}^{(i)} - \hat{\mathbf{u}}^{(i)})). \end{aligned}$$

Letting  $\nabla F(\hat{\mathbf{z}}) = \mathbf{0}$ , we have

$$(\hat{\mathbf{D}}^\top \hat{\mathbf{D}} + \rho \mathbf{I})_{mn \times mn} \hat{\mathbf{z}} = \hat{\mathbf{D}}^\top \hat{\mathbf{x}} + \rho(\hat{\mathbf{y}}^{(i)} - \hat{\mathbf{u}}^{(i)}).$$

Therefore, we obtain the solution

$$\hat{\mathbf{z}}^{(i+1)} = (\hat{\mathbf{D}}^\top \hat{\mathbf{D}} + \rho \mathbf{I})^{-1} (\hat{\mathbf{D}}^\top \hat{\mathbf{x}} + \rho(\hat{\mathbf{y}}^{(i)} - \hat{\mathbf{u}}^{(i)})).$$

## Solving minimization problem (13<sub>2</sub>)

---

The way to solve minimization problem (13<sub>2</sub>) is similar to that for solving problem (6<sub>2</sub>).

Finally, we obtain the ADMM iterative scheme as follows:

$$\hat{\mathbf{z}}^{(i+1)} = (\hat{\mathbf{D}}^\top \hat{\mathbf{D}} + \rho \mathbf{I})^{-1} (\hat{\mathbf{D}}^\top \hat{\mathbf{x}} + \rho (\hat{\mathbf{y}}^{(i)} - \hat{\mathbf{u}}^{(i)})), \quad (14_1)$$

$$\mathbf{y}^{(i+1)} = \mathcal{S}_{\lambda/\rho}(\mathcal{F}^{-1}(\hat{\mathbf{z}}^{(i+1)}) + \mathbf{u}^{(i)}), \quad (14_2)$$

$$\mathbf{u}^{(i+1)} = \mathbf{u}^{(i)} + \mathcal{F}^{-1}(\hat{\mathbf{z}}^{(i+1)}) - \mathbf{y}^{(i+1)}. \quad (14_3)$$

Next, we will introduce the *Sherman-Morrison formula* which can be applied to solve  $\hat{\mathbf{z}}^{(i+1)}$  in a more efficient way.

## The Sherman-Morrison formula: a special case

Let  $A$  and  $B$  be two  $n \times n$  matrices. In general,  $(A + B)$  is not invertible, even though  $A$  is invertible. However, if  $A$  is invertible and  $B$  has some certain structure, then  $(A + B)^{-1}$  exists.

**A special case of the Sherman-Morrison formula:** *Let  $I$  be the  $n \times n$  identity matrix and  $u, v$  be two given vectors in  $\mathbb{C}^n$ . If  $1 + v^\top u \neq 0$ , then  $I + uv^\top$  is invertible and*

$$(I + uv^\top)^{-1} = I - \frac{uv^\top}{1 + v^\top u}.$$

*Proof:* We check that

$$\begin{aligned} (I + uv^\top) \left( I - \frac{uv^\top}{1 + v^\top u} \right) &= I - \frac{uv^\top}{1 + v^\top u} + uv^\top - \frac{uv^\top uv^\top}{1 + v^\top u} \\ &= I + \frac{-uv^\top + uv^\top + v^\top u uv^\top}{1 + v^\top u} - \frac{uv^\top uv^\top}{1 + v^\top u} \\ &= I + \frac{v^\top u uv^\top}{1 + v^\top u} - \frac{uv^\top uv^\top}{1 + v^\top u} = I. \quad (v^\top u : \text{scalar}) \quad \square \end{aligned}$$

## How to derive the inverse?

Given  $\mathbf{b} \in \mathbb{C}^n$ , we consider the linear system  $(\mathbf{I} + \mathbf{u}\mathbf{v}^\top)\mathbf{x} = \mathbf{b}$ . Assume that  $\mathbf{I} + \mathbf{u}\mathbf{v}^\top$  is invertible. Then the unique solution  $\mathbf{x}$  exists.

Let  $k = \mathbf{v}^\top \mathbf{x} \in \mathbb{C}$ . Then  $\mathbf{x} + k\mathbf{u} = \mathbf{b} \Rightarrow \mathbf{v}^\top \mathbf{x} + k\mathbf{v}^\top \mathbf{u} = \mathbf{v}^\top \mathbf{b}$   
 $\Rightarrow k + k(\mathbf{v}^\top \mathbf{u}) = \mathbf{v}^\top \mathbf{b}$ , which implies

$$k = \frac{\mathbf{v}^\top \mathbf{b}}{1 + \mathbf{v}^\top \mathbf{u}}, \quad \text{if } 1 + \mathbf{v}^\top \mathbf{u} \neq 0.$$

Therefore, we know that

$$\mathbf{x} = \mathbf{b} - k\mathbf{u} = \mathbf{b} - \frac{\mathbf{v}^\top \mathbf{b}}{1 + \mathbf{v}^\top \mathbf{u}} \mathbf{u} = \mathbf{b} - \frac{\mathbf{u}\mathbf{v}^\top}{1 + \mathbf{v}^\top \mathbf{u}} \mathbf{b} = \underbrace{\left( \mathbf{I} - \frac{\mathbf{u}\mathbf{v}^\top}{1 + \mathbf{v}^\top \mathbf{u}} \right)}_{\text{inverse of } \mathbf{I} + \mathbf{u}\mathbf{v}^\top} \mathbf{b}.$$

**The Sherman-Morrison formula:** *Suppose that  $\mathbf{A} \in \mathbb{C}^{n \times n}$  is an invertible matrix and  $\mathbf{u}, \mathbf{v} \in \mathbb{C}^n$ . Then  $\mathbf{A} + \mathbf{u}\mathbf{v}^\top$  is invertible if and only if  $1 + \mathbf{v}^\top \mathbf{A}^{-1} \mathbf{u} \neq 0$ . In this case, we have*

$$(\mathbf{A} + \mathbf{u}\mathbf{v}^\top)^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1} \mathbf{u}\mathbf{v}^\top \mathbf{A}^{-1}}{1 + \mathbf{v}^\top \mathbf{A}^{-1} \mathbf{u}}.$$

## How to compute (14<sub>1</sub>)?

Recall that

$$(\widehat{\mathbf{D}}^\top \widehat{\mathbf{D}} + \rho \mathbf{I}) \widehat{\mathbf{z}}^{(i+1)} = (\widehat{\mathbf{D}}^\top \widehat{\mathbf{x}} + \rho(\widehat{\mathbf{y}}^{(i)} - \widehat{\mathbf{u}}^{(i)})), \quad (14_1)$$

where matrix  $\widehat{\mathbf{D}}$  has the following structure:

$$\begin{aligned} \widehat{\mathbf{D}} &= [\widehat{\mathbf{D}}_1, \widehat{\mathbf{D}}_2, \dots, \widehat{\mathbf{D}}_n]_{m \times mn} \\ &= \begin{bmatrix} \widehat{d'_{1,1}} & 0 & \cdots & 0 & \widehat{d'_{2,1}} & 0 & \cdots & 0 & \cdots \\ 0 & \widehat{d'_{1,2}} & \ddots & \vdots & 0 & \widehat{d'_{2,2}} & \ddots & \vdots & \cdots \\ \vdots & \ddots & \ddots & 0 & \vdots & \ddots & \ddots & 0 & \cdots \\ 0 & \cdots & 0 & \widehat{d'_{1,m}} & 0 & \cdots & 0 & \widehat{d'_{2,m}} & \cdots \end{bmatrix}, \end{aligned}$$

where

$$\widehat{\mathbf{D}}_j = \text{diag}(\widehat{d'_j}) \quad (m \times m \text{ diagonal matrix}).$$

## The structure of matrix $\widehat{\mathbf{D}}^\top \widehat{\mathbf{D}} + \rho \mathbf{I}$

Note that

$$\begin{aligned} \widehat{\mathbf{D}}^\top \widehat{\mathbf{D}} + \rho \mathbf{I} &= \begin{bmatrix} \widehat{\mathbf{D}}_1^\top \\ \widehat{\mathbf{D}}_2^\top \\ \vdots \\ \widehat{\mathbf{D}}_n^\top \end{bmatrix} [\widehat{\mathbf{D}}_1, \widehat{\mathbf{D}}_2, \dots, \widehat{\mathbf{D}}_n] + \rho \mathbf{I} \\ &= \begin{bmatrix} \widehat{\mathbf{D}}_1^\top \widehat{\mathbf{D}}_1 + \rho \mathbf{I}_m & \widehat{\mathbf{D}}_1^\top \widehat{\mathbf{D}}_2 & \cdots & \widehat{\mathbf{D}}_1^\top \widehat{\mathbf{D}}_n \\ \widehat{\mathbf{D}}_2^\top \widehat{\mathbf{D}}_1 & \widehat{\mathbf{D}}_2^\top \widehat{\mathbf{D}}_2 + \rho \mathbf{I}_m & \cdots & \widehat{\mathbf{D}}_2^\top \widehat{\mathbf{D}}_n \\ \vdots & \vdots & \ddots & \vdots \\ \widehat{\mathbf{D}}_n^\top \widehat{\mathbf{D}}_1 & \widehat{\mathbf{D}}_n^\top \widehat{\mathbf{D}}_2 & \cdots & \widehat{\mathbf{D}}_n^\top \widehat{\mathbf{D}}_n + \rho \mathbf{I}_m \end{bmatrix}_{mn \times mn} \end{aligned}$$

*By re-ordering the equations, the  $mn \times mn$  system  $(\widehat{\mathbf{D}}^\top \widehat{\mathbf{D}} + \rho \mathbf{I}) \widehat{\mathbf{z}}^{(i+1)} = \mathbf{R}$  can be replaced by  $m$  independent linear systems of size  $n \times n$ , each of which consists of a rank one component plus a diagonal component, then solved by the Sherman-Morrison formula, see [Wohlberg 2016, Appendix A].*

## Convolutional sparse dictionary learning problem

We now consider the convolutional sparse dictionary learning problem, where the dictionary  $\mathbf{D}$  is unknown and needed to be sought together with the convolutional sparse solution.

**Convolutional SDL problem:** Let  $\{\mathbf{x}_i\}_{i=1}^N \subset \mathbb{R}^m$  be a given dataset of signals. We seek a dictionary matrix  $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_M] \in \mathbb{R}^{\ell \times M}$  and the coefficient matrices  $\{\mathbf{Z}_i\}_{i=1}^N \subset \mathbb{R}^{k \times M}$  with  $\mathbf{Z}_i = [z_{1,i}, z_{2,i}, \dots, z_{M,i}]$  and  $m = \ell + k - 1$  such that  $\mathbf{D}$  and  $\{\mathbf{Z}_i\}_{i=1}^N$  solve the following minimization problem:

$$\min_{\{\mathbf{d}_j\}_{j=1}^M, \{\mathbf{z}_{j,i}\}_{j=1, i=1}^{M,N}} \left( \frac{1}{2} \sum_{i=1}^N \left\| \mathbf{x}_i - \sum_{j=1}^M \mathbf{d}_j * \mathbf{z}_{j,i} \right\|_2^2 + \lambda \sum_{i=1}^N \sum_{j=1}^M \|\mathbf{z}_{j,i}\|_1 \right)$$

subject to  $\|\mathbf{d}_j\|_2 \leq 1 \quad \forall j = 1, 2, \dots, M,$

where  $\lambda > 0$  is a given penalty parameter.

## Toeplitz matrix

- Define  $\tilde{\mathbf{D}} = [\mathbf{D}_1 \quad \mathbf{D}_2 \quad \cdots \quad \mathbf{D}_M]$  with each  $\mathbf{D}_j$  is the *Toeplitz matrix* defined with respect to  $\mathbf{d}_j$  as before.

Define  $\mathbf{z}_i = [\mathbf{z}_{1,i}^\top \quad \mathbf{z}_{2,i}^\top \quad \cdots \quad \mathbf{z}_{M,i}^\top]^\top$  for  $i = 1, 2, \dots, N$ , where each  $\mathbf{z}_i$  is the coefficient vector with respect to the data  $\mathbf{x}_i$ .

Define  $\mathbf{Z} = [\mathbf{z}_1 \quad \mathbf{z}_2 \quad \cdots \quad \mathbf{z}_N]$  and  $\mathbf{X} = [\mathbf{x}_1 \quad \mathbf{x}_2 \quad \cdots \quad \mathbf{x}_N]$ .

- The convolutional SDL problem can be simplified as

$$\begin{aligned} \min_{\tilde{\mathbf{D}}, \mathbf{Z}} \left( \frac{1}{2} \|\mathbf{X} - \tilde{\mathbf{D}}\mathbf{Z}\|_F^2 + \lambda \|\mathbf{Z}\|_{1,1} \right) \\ \text{subject to } \|\mathbf{d}_j\|_2 \leq 1 \quad \forall j = 1, 2, \dots, M, \end{aligned}$$

where  $\|\mathbf{Z}\|_{1,1}$  is defined as before.

## How to solve the convolutional SDL problem?

---

Though we can still use the ADMM iterative scheme to solve

$$\begin{aligned} \min_{\tilde{D}, Z} & \left( \frac{1}{2} \|X - \tilde{D}Z\|_F^2 + \lambda \|Z\|_{1,1} \right) \\ & \text{subject to } \|d_j\|_2 \leq 1 \quad \forall j = 1, 2, \dots, M, \end{aligned}$$

the sizes of the involved matrices are too large. Thus, we will use the DFT and the Sherman-Morrison formula to deal with this problem. The steps are similar to the CSR problem, but more complicated.

Recall the convolutional SDL problem:

$$\begin{aligned} \min_{\{d_j\}_{j=1}^M, \{z_{j,i}\}_{j=1,i=1}^{M,N}} & \left( \frac{1}{2} \sum_{i=1}^N \|x_i - \sum_{j=1}^M d_j * z_{j,i}\|_2^2 + \lambda \sum_{i=1}^N \sum_{j=1}^M \|z_{j,i}\|_1 \right) \\ & \text{subject to } \|d_j\|_2 \leq 1 \quad \forall j = 1, 2, \dots, M. \end{aligned}$$

For solving this problem, we split it into two parts.

## Step 1: Solving the coefficient $\mathbf{Z}$

For solving the convolutional SDL problem, we first give an initial dictionary  $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_M]$  to solve coefficient  $\mathbf{Z}$  and then further use ADMM algorithm to split this problem into three subproblems:

$$\widehat{\mathbf{Z}}^{(i+1)} = \arg \min_{\widehat{\mathbf{Z}}} \left( \frac{1}{2} \sum_{i=1}^N \|\widehat{\mathbf{x}}_i - \sum_{j=1}^M \widehat{\mathbf{d}}'_j \odot \widehat{\mathbf{z}}'_{j,i}\|_2^2 + \frac{\rho}{2} \sum_{i=1}^N \sum_{j=1}^M \|\widehat{\mathbf{z}}'_{j,i} - \mathbf{y}'_{j,i}^{(i)} + \mathbf{u}'_{j,i}^{(i)}\|_2^2 \right)$$

$$\mathbf{Y}^{(i+1)} = \arg \min_{\mathbf{Y}} \left( \lambda \sum_{i=1}^N \sum_{j=1}^M \|\mathbf{y}_{j,i}\|_1 + \frac{\rho}{2} \sum_{i=1}^N \sum_{j=1}^M \|\mathcal{F}^{-1}(\widehat{\mathbf{z}}'_{j,i}) - \mathbf{y}_{j,i} + \mathbf{u}'_{j,i}\|_2^2 \right),$$

$$\mathbf{U}^{(i+1)} = \mathbf{U}^{(i)} + \mathcal{F}^{-1}(\widehat{\mathbf{Z}}^{(i+1)}) - \mathbf{Y}^{(i+1)},$$

with

$$\mathcal{F}^{-1}(\widehat{\mathbf{Z}}) = \begin{bmatrix} \mathcal{F}^{-1}(\widehat{\mathbf{z}}'_{1,1}) & \cdots & \mathcal{F}^{-1}(\widehat{\mathbf{z}}'_{1,M}) \\ \vdots & \ddots & \vdots \\ \mathcal{F}^{-1}(\widehat{\mathbf{z}}'_{N,1}) & \cdots & \mathcal{F}^{-1}(\widehat{\mathbf{z}}'_{N,M}) \end{bmatrix}.$$

## Rewritten in a compact form

For convenience, we can rewrite these subproblems as follows:

$$\widehat{\mathbf{Z}}^{(i+1)} = \arg \min_{\widehat{\mathbf{Z}}} \left( \frac{1}{2} \|\widehat{\mathbf{X}} - \widehat{\mathbf{D}}\widehat{\mathbf{Z}}\|_F^2 + \frac{\rho}{2} \|\widehat{\mathbf{Z}} - \widehat{\mathbf{Y}}^{(i)} + \widehat{\mathbf{U}}^{(i)}\|_F^2 \right),$$

$$\mathbf{Y}^{(i+1)} = \arg \min_{\mathbf{Y}} \left( \lambda \|\mathbf{Y}\|_{1,1} + \frac{\rho}{2} \|\mathcal{F}^{-1}(\widehat{\mathbf{Z}}^{(i+1)}) - \widehat{\mathbf{Y}} + \widehat{\mathbf{U}}^{(i)}\|_F^2 \right),$$

$$\mathbf{U}^{(i+1)} = \mathbf{U}^{(i)} + \mathcal{F}^{-1}(\widehat{\mathbf{Z}}^{(i+1)}) - \mathbf{Y}^{(i+1)},$$

with

$$\widehat{\mathbf{X}} = [\widehat{\mathbf{x}}_1, \widehat{\mathbf{x}}_2, \dots, \widehat{\mathbf{x}}_N], \quad \widehat{\mathbf{D}} = [\widehat{\mathbf{D}}_1, \widehat{\mathbf{D}}_2, \dots, \widehat{\mathbf{D}}_M],$$

and

$$\mathbf{Y} = \begin{bmatrix} \mathbf{y}'_{1,1} & \cdots & \mathbf{y}'_{1,M} \\ \vdots & \ddots & \vdots \\ \mathbf{y}'_{N,1} & \cdots & \mathbf{y}'_{N,M} \end{bmatrix}, \quad \mathbf{U} = \begin{bmatrix} \mathbf{u}'_{1,1} & \cdots & \mathbf{u}'_{1,M} \\ \vdots & \ddots & \vdots \\ \mathbf{u}'_{N,1} & \cdots & \mathbf{u}'_{N,M} \end{bmatrix}.$$

Using the similar ways as that for solving CSR problem, we can solve the above subproblems.

## Step 2: Solving the dictionary $D$

---

Recall the convolutional SDL problem:

$$\min_{\{\mathbf{d}_j\}_{j=1}^M, \{z_{j,i}\}_{j=1,i=1}^{M,N}} \left( \frac{1}{2} \sum_{i=1}^N \|\mathbf{x}_i - \sum_{j=1}^M \mathbf{d}_j * z_{j,i}\|_2^2 + \lambda \sum_{i=1}^N \sum_{j=1}^M \|z_{j,i}\|_1 \right)$$

subject to  $\|\mathbf{d}_j\|_2 \leq 1 \quad \forall j = 1, 2, \dots, M.$

When the coefficient  $\mathbf{Z}$  is obtained, the blue term is a given number. Solving the dictionary  $D$  is equivalent to solve

$$\min_{\{\mathbf{d}_j\}_{j=1}^M} \frac{1}{2} \sum_{i=1}^N \|\mathbf{x}_i - \sum_{j=1}^M \mathbf{d}_j * z_{j,i}\|_2^2 \quad \text{subject to } \|\mathbf{d}_j\|_2 \leq 1, \quad \forall j = 1, 2, \dots, M.$$

## Using ADMM algorithm to solve Step 2

We use the ADMM algorithm to solve the above problem:

$$\mathbf{D}^{(i+1)} = \arg \min_{\mathbf{D}} \left( \frac{1}{2} \sum_{i=1}^N \|x_i - \sum_{j=1}^M \mathbf{d}_j * z_j\|_2^2 + \frac{\rho}{2} \sum_{j=1}^M \|\mathbf{d}_j - \mathbf{g}_j^{(i)} + \mathbf{h}_j^{(i)}\|_2^2 \right),$$

$$\mathbf{G}^{(i+1)} = \text{proj}_{\mathbf{g}(\mathbf{G})} \{\mathbf{D}^{(i+1)}\},$$

$$\mathbf{H}^{(i+1)} = \mathbf{H}^{(i)} + \mathbf{D}^{(i+1)} - \mathbf{G}^{(i+1)},$$

and then use the Fourier transform and similar ways as before,

$$\widehat{\mathbf{D}}^{(i+1)} = \arg \min_{\widehat{\mathbf{D}}} \left( \frac{1}{2} \sum_{i=1}^N \|\widehat{x}_i - \sum_{j=1}^M \widehat{\mathbf{d}}'_j \odot \widehat{z}_j\|_2^2 + \frac{\rho}{2} \sum_{j=1}^M \|\widehat{\mathbf{d}}'_j - \widehat{\mathbf{g}}_j^{(i)} + \widehat{\mathbf{h}}_j^{(i)}\|_2^2 \right),$$

$$\mathbf{G}^{(i+1)} = \text{proj}_{\mathbf{g}(\mathbf{G})} \{\mathcal{F}^{-1}(\widehat{\mathbf{D}}^{(i+1)})\},$$

$$\mathbf{H}^{(i+1)} = \mathbf{H}^{(i)} + \mathcal{F}^{-1}(\widehat{\mathbf{D}}^{(i+1)}) - \mathbf{G}^{(i+1)},$$

where

$$\mathcal{F}^{-1}(\widehat{\mathbf{D}}) = [\mathcal{F}^{-1}(\widehat{\mathbf{d}}'_1), \mathcal{F}^{-1}(\widehat{\mathbf{d}}'_2), \dots, \mathcal{F}^{-1}(\widehat{\mathbf{d}}'_M)].$$

## References

---

- [1] H. Bristow, A. Eriksson, and S. Lucey, Fast convolutional sparse coding, *2013 IEEE CVPR*, pp. 391-398.
- [2] F. Heide, W. Heidrich, and G. Wetzstein, Fast and flexible convolutional sparse coding, *2015 IEEE CVPR*, pp. 5135-5143.
- [3] H. Sreter and R. Giryes, Learned convolutional sparse coding, *2018 IEEE ICASSP*, pp. 2191-2195.
- [4] B. Wohlberg, Efficient convolutional sparse coding, *2014 IEEE ICASSP*, pp. 7173-7177.
- [5] B. Wohlberg, Efficient algorithms for convolutional sparse representations, *IEEE Transactions on Image Processing*, 25 (2016), pp. 301-315.
- [6] J. Xiong, P. Richtarik, and W. Heidrich, Stochastic convolutional sparse coding, *Vision, Modeling, and Visualization*, 2019.