# Total variation norm regularized least square

i.e. minimize $\|\mathbf{Ax}-\mathbf{b}\|_2^2+\|\mathbf{Dx}\|_1$ where $\|\mathbf{Dx}\|_1$ is not proximable

Andersen Ang

Mathématique et recherche opérationnelle
UMONS, Belgium

manshun.ang@umons.ac.be Homepage: angms.science

First draft : November, 22, 2019
Last update : November 24, 2019

# Total variation norm regularization

Given $\mathbf{A} \in \mathbb{R}^{m \times n}, \mathbf{b} \in \mathbb{R}^m$, a scalar $\lambda \geq 0$, find the vector $\mathbf{x} \in \mathbb{R}^n$ by solving

$$(\mathcal{P}) : \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda\|\mathbf{D}\mathbf{x}\|_1$$

where $\mathbf{D}$ is the difference operator

$$\mathbf{D} = \begin{bmatrix} 1 & -1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & -1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & 1 & -1 \end{bmatrix} \in \mathbb{R}^{(n-1)\times n}$$

- $\|\mathbf{D}\mathbf{x}\|_1$ is the total variation (TV) norm
- $\|\mathbf{D}\mathbf{x}\|_1 = \sum_{i=1}^{n-1} \left|x(i) - x(i+1)\right|$, it measure the difference between coordinate : i.e. it detect sharp changes
- TV norm regularization has many applications
- Special case : for $m = n$ and $\mathbf{A} = \mathbf{I}_n$, $(\mathcal{P})$ is edge-preserving de-noising

# Proximal gradient algorithm

Cost function in $(\mathcal{P})$ is in the form $\min_{\mathbf{x}} f(\mathbf{x}) + \lambda g(\mathbf{x})$ where $f$ is smooth but $g$ is not smooth.
Standard first-order approach to solve this kind of problem is the proximal gradient

$$\mathbf{x}^{[k+1]} = \text{prox}_{\lambda g}\Big(\mathbf{x}^{[k]} - \alpha \nabla f(\mathbf{x}^{[k]})\Big),$$

where $\alpha$ is step size for the "forward step", and the proximal operator $\text{prox}_{\lambda g}(\cdot)$ is the"backward step"

$$\text{prox}_{\lambda g}(\mathbf{z}) = \arg\min_{\mathbf{x}} \left(g(\mathbf{x}) + \frac{1}{2\lambda}\|\mathbf{x} - \mathbf{z}\|_2^2\right).$$

See here for the convergence property of proximal gradient algorithm.

# No close form expression for the proximal operator

$$(\mathcal{P}) \;:\; \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 + \lambda\|\mathbf{D}\mathbf{x}\|_1$$

using proximal gradient requires to solve the sub-problem

$$\begin{aligned}
\mathbf{x}^{[k+1]} &= \text{prox}_{\lambda g}(\mathbf{z}) \\
&= \arg\min_{\mathbf{x}} \left( \|\mathbf{D}\mathbf{x}\|_1 + \frac{1}{2\lambda}\|\mathbf{x}-\mathbf{z}\|_2^2 \right) \\
&= \arg\min_{\mathbf{x}} \left( \|\mathbf{D}\mathbf{x}\|_1 + \frac{1}{2\lambda}\|\mathbf{x}-\left(\mathbf{x}^{[k]} - \alpha\nabla f(\mathbf{x}^{[k]})\right)\|_2^2 \right)
\end{aligned}$$

with no analytic close form solution.

A way to solve this sub-problem is *Majorization Minimization* (MM).
See here for more background.

A main drawback of MM : slow.

# The majorizer of absolute value and $\|\mathbf{x}\|_1$

The key in MM is the majorizer.
It can be shown absolute value function has a tight quadratic upper bound

$$|x| \leq ax^2 + bx + c, \quad a = \frac{1}{2|\bar{x}|}, b = 0, c = \frac{1}{2}|\bar{x}| \tag{1}$$

(1) can be derived by considering the MM criterion.

As $\|\mathbf{x}\|_1 =$ sum of absolute value of components, we can make use of (1) to each component of $\mathbf{x}$ to get the majorizer.

$$\|\mathbf{x}\|_1 = \sum_{i=1}^{n} |x_i| \leq \sum_{i=1}^{n} \left( \frac{1}{2|\bar{x}_i|} x_i^2 + \frac{1}{2}|\bar{x}_i| \right) = \frac{1}{2}\mathbf{x}^\top \Lambda^{-1} \mathbf{x} + \frac{1}{2}\|\bar{\mathbf{x}}\|_1$$

where $\Lambda = \text{Diag}\Big(|\bar{x}_1|, \ldots, |\bar{x}_n|\Big)$, denoted as $\text{Diag}(|\bar{\mathbf{x}}|)$.

# The majorizer of $\|\mathbf{Dx}\|_1$

For $\|\mathbf{Dx}\|_1$, let $\mathbf{y} = \mathbf{Dx}$ and apply $\|\mathbf{x}\|_1 \leq \frac{1}{2}\mathbf{x}^\top \Lambda^{-1}\mathbf{x} + \frac{1}{2}\|\bar{\mathbf{x}}\|_1$ on $\mathbf{y}$ :

$$\|\mathbf{Dx}\|_1 \leq \frac{1}{2}\mathbf{x}^\top \mathbf{D}^\top \Lambda^{-1}\mathbf{Dx} + \frac{1}{2}\|\mathbf{D}\bar{\mathbf{x}}\|_1 \tag{2}$$

where $\Lambda$ =Diag($|\mathbf{D}\bar{\mathbf{x}}|$).

(2) is the core of the MM approach to solve $(\mathcal{P})$.

Note that the term $\frac{1}{2}\|\mathbf{D}\bar{\mathbf{x}}\|_1$ is independent of $\mathbf{x}$, this term is a constant and can be ignored in the optimization process

$$\arg\min_{\mathbf{x}} \frac{1}{2}\mathbf{x}^\top \mathbf{D}^\top \Lambda^{-1}\mathbf{Dx} + \frac{1}{2}\|\mathbf{D}\bar{\mathbf{x}}\|_1 = \arg\min_{\mathbf{x}} \frac{1}{2}\mathbf{x}^\top \mathbf{D}^\top \Lambda^{-1}\mathbf{Dx}.$$

# Solving problem $(\mathcal{P})$ by MM

We solve $(\mathcal{P})$ indirectly by solving

$$(\mathcal{P}') \ : \ \arg\min_{\mathbf{x}} F(\mathbf{x}) = \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 + \frac{\lambda}{2}\mathbf{x}^\top \mathbf{D}^\top \Lambda^{-1}\mathbf{D}\mathbf{x},$$

which is to minimize a quadratic function.
To minimize $F$, we can consider the first order optimality condition $\nabla F(\mathbf{x}) = 0$, which gives

$$\mathbf{x} = (\mathbf{A}^\top\mathbf{A} + \lambda\mathbf{D}^\top\Lambda^{-1}\mathbf{D})^{-1}\mathbf{A}^\top\mathbf{b}. \tag{3}$$

To improve the numerical stability of (3), we make use of the matrix inversion lemma

$$(\mathbf{X}+\mathbf{U}\mathbf{Y}\mathbf{V})^{-1} = \mathbf{X}^{-1} - \mathbf{X}^{-1}\mathbf{U}(\mathbf{Y}^{-1}+\mathbf{V}\mathbf{X}^{-1}\mathbf{U})^{-1}\mathbf{V}\mathbf{X}^{-1}$$

Hence (3) becomes

$$\mathbf{x} = \Big((\mathbf{A}^\top\mathbf{A})^{-1} - (\mathbf{A}^\top\mathbf{A})^{-1}\mathbf{D}^\top(\frac{1}{\lambda}\Lambda + \mathbf{D}(\mathbf{A}^\top\mathbf{A})^{-1}\mathbf{D}^\top)\mathbf{D}(\mathbf{A}^\top\mathbf{A})^{-1}\Big)\mathbf{A}^\top\mathbf{b}$$

# MM-based algorithm for solving problem $(\mathcal{P})$

The flow of the algorithm is

- Pick initial guess $\mathbf{x}_0$
- For $k = 1, 2, \ldots$
  - $\Lambda_k = \text{Diag}\big(|\mathbf{D}\mathbf{x}_k|\big)$
  - Update $\mathbf{x}^{[k+1]}$ as follows :
    1. direct $(\mathbf{A}^\top\mathbf{A} + \lambda\mathbf{D}^\top\Lambda^{-1}\mathbf{D})^{-1}\mathbf{A}^\top\mathbf{b}$.
    2. indirect

$$\Big((\mathbf{A}^\top\mathbf{A})^{-1} - (\mathbf{A}^\top\mathbf{A})^{-1}\mathbf{D}^\top\big(\frac{1}{\lambda}\Lambda_k + \mathbf{D}(\mathbf{A}^\top\mathbf{A})^{-1}\mathbf{D}^\top\big)\mathbf{D}(\mathbf{A}^\top\mathbf{A})^{-1}\Big)\mathbf{A}^\top\mathbf{b}$$

Remarks : constant terms such as $(\mathbf{A}^\top\mathbf{A})^{-1}$, $\mathbf{A}^\top\mathbf{b}$, $(\mathbf{A}^\top\mathbf{A})^{-1}\mathbf{A}^\top\mathbf{b}$ should be pre-computed to improve efficiency.

# MM-based algorithm for solving problem $(\mathcal{P})$ : special case

Consider the special case $m = n$ and $\mathbf{A} = \mathbf{I}_n$.

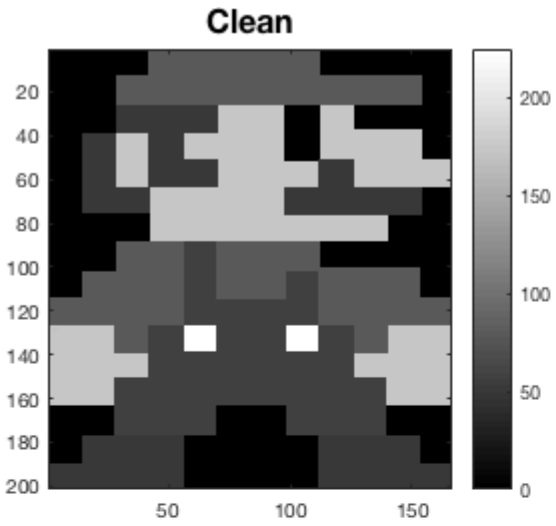
The flow of the algorithm is

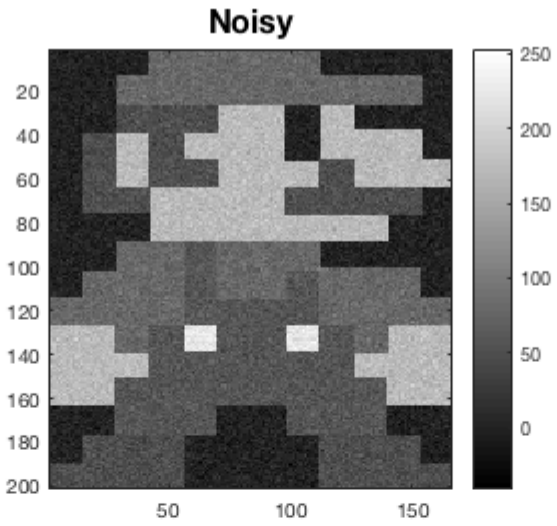
- Pick initial guess $\mathbf{x}_0$
- For $k = 1, 2, \ldots$
  - $\Lambda_k = \text{Diag}\big(|\mathbf{D}\mathbf{x}_k|\big)$
  - Update $\mathbf{x}^{[k+1]}$ as follows :
    1. direct $(\mathbf{I} + \lambda \mathbf{D}^\top \Lambda^{-1} \mathbf{D})^{-1}\mathbf{b}$.
    2. indirect $\Big(\mathbf{I} - \mathbf{D}^\top (\frac{1}{\lambda}\Lambda_k + \mathbf{D}\mathbf{D}^\top)\mathbf{D}\Big)\mathbf{b}$

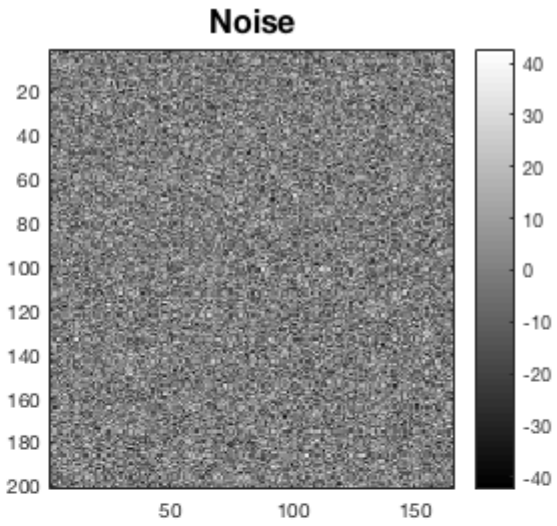
Remarks

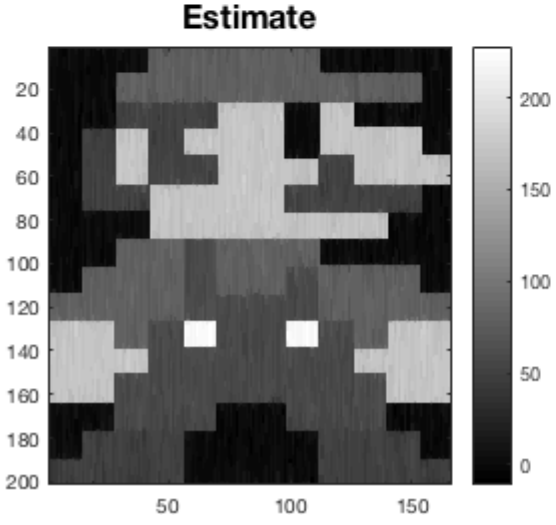
- again constant terms can be pre-computed to improve efficiency. e.g., for the indirect update, update $\mathbf{x}^{[k+1]}$ as $\underbrace{\Big(\mathbf{I} - (\mathbf{D}^\top \mathbf{D})^2\Big)\mathbf{b}}_{\mathbf{C}_1} - \underbrace{\frac{1}{\lambda}\mathbf{D}^\top}_{\mathbf{C}_2} \Lambda_k \underbrace{\mathbf{D}\mathbf{b}}_{\mathbf{c}_3}$, and pre-compute $\mathbf{C}_1, \mathbf{C}_2, \mathbf{c}_3$.
- $\mathbf{I}$ is diagonal and $\mathbf{D}^\top \mathbf{D}$ is banded, special data structure can be used in implementation. e.g. in MATLAB, create $\mathbf{I}$ as `speye(n)` instead of `eye(n)`.

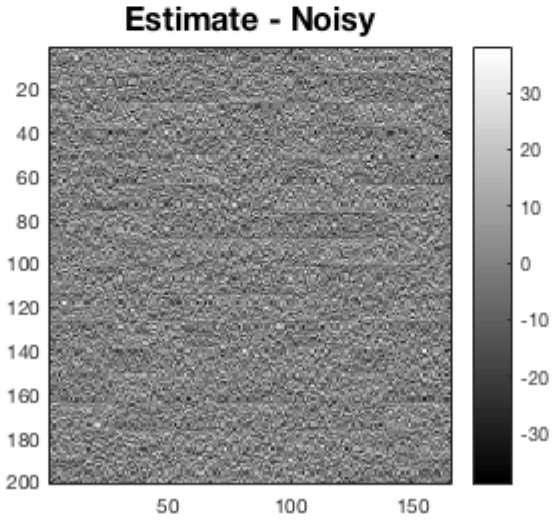
# Example

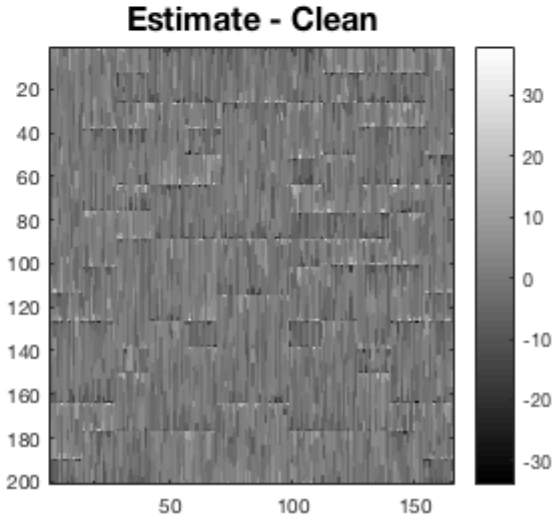

Clean
20
40
60
80
100
120
140
160
180
200
50
100
150
200
150
100
50
0


Noisy
20
40
60
80
100
120
140
160
180
200
50
100
150
250
200
150
100
50
0


Noise
20
40
60
80
100
120
140
160
180
200
50
100
150
40
30
20
10
0
-10
-20
-30
-40

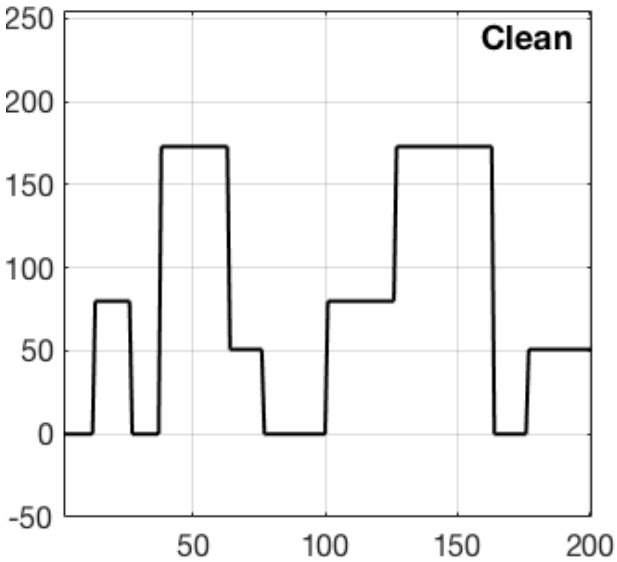

Estimate
20
40
60
80
100
120
140
160
180
200
50
100
150
200
150
100
50
0

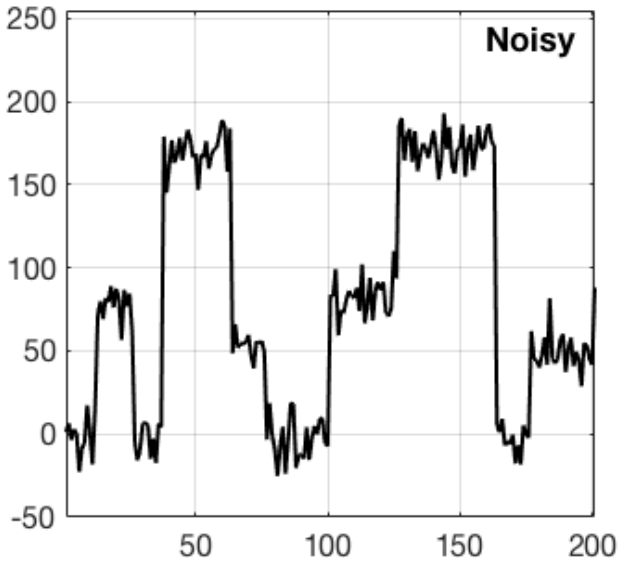

Estimate - Noisy
20
40
60
80
100
120
140
160
180
200
50
100
150
30
20
10
0
-10
-20
-30

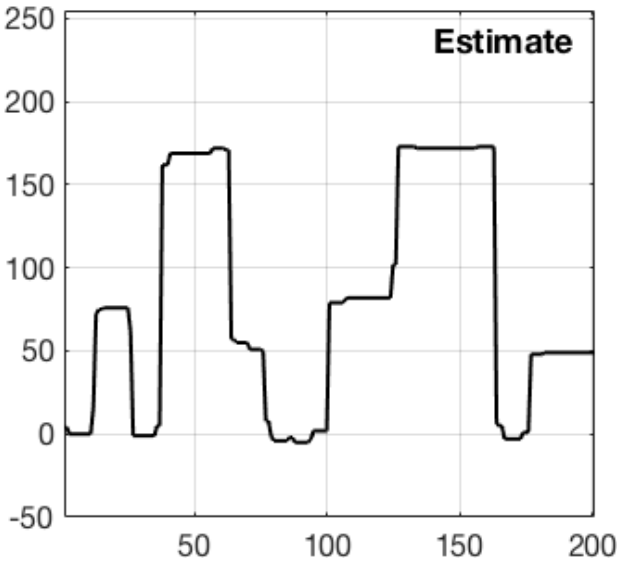

Estimate - Clean
20
40
60
80
100
120
140
160
180
200
50
100
150
30
20
10
0
-10
-20
-30

# Example

One column of the image



# Last page - summary

Solving $\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 + \lambda\|\mathbf{D}\mathbf{x}\|_1$ by MM

- $|x| \leq \frac{1}{2|\bar{x}|} + \frac{1}{2}|\bar{x}|$
- $\|\mathbf{x}\|_1 \leq \frac{1}{2}\mathbf{x}^\top \Big(\text{Diag}(|\bar{\mathbf{x}}|)\Big)^{-1}\mathbf{x} + \frac{1}{2}\|\bar{\mathbf{x}}\|_1$
- $\|\mathbf{D}\mathbf{x}\|_1 \leq \frac{1}{2}\mathbf{D}^\top\mathbf{x}^\top \Big(\text{Diag}(|\mathbf{D}\bar{\mathbf{x}}|)\Big)^{-1}\mathbf{D}\mathbf{x} + \frac{1}{2}\|\mathbf{D}\bar{\mathbf{x}}\|_1$
- $\mathbf{x} = (\mathbf{A}^\top\mathbf{A} + \lambda\mathbf{D}^\top\Lambda^{-1}\mathbf{D})^{-1}\mathbf{A}\mathbf{b}$ and the improved version
- Special case when $\mathbf{A} = \mathbf{I}_n$
- Application in image de-noising for edgy image (image with many sharp corners)

End of document