Lecture X. Enhancing Sparse Signal Recovery

Hojin Kim

Department of Radiation Oncology,
Yonsei Cancer Center, Severance Hospital,
Yonsei University College of Medicine

Severance

In Lecture IX

- Review of Transform Sparsity
  - Sparse signal recovery for actual images
  - Transform sparsity with total-variation minimization (ADMM, Chambolle-Pock algorithms)
  - Various transforms available for image reconstruction and denoising

- Today, Let's take a look at **how to enhance sparse signal recovery**

Enhancing Time Efficiency

Accelerating Strategy

Accelearting Strategy

- Unconstrained Minimization
  - f is **convex** and **differentiable**
  - ∇f is **Lipschitz constant *L*** $\|\nabla f(x) - \nabla f(y)\|_2 \le L\|x-y\|_2$

$$t \le \frac{1}{L}: \qquad f(x_k) - f^* \le \frac{\|x_0 - x^*\|_2^2}{2tk}$$

Linear convergence

- Method to make the convergence faster

$$x_{k+1} = x_k - t_k \nabla f(x_k) + \boxed{s_k(x_k - x_{k-1})}$$

Utilizing 'previous iterates'

FISTA -- Accelerating ISTA

• FISTA (Fast Iterative Soft-Thresholding Algorithm)

$$x_{k+1} = y_k - t_k \nabla f(y_k)$$
$$y_{k+1} = x_{k+1} + \frac{k}{k+3}(x_{k+1} - x_k)$$
$$\Rightarrow \quad f(x_k) - f^* \le \frac{2\|x_0 - x^*\|_2^2}{t(k+1)^2}$$

Quadratic convergence

$$\underset{x}{\text{minimize}} \;\; f(x) = \|Ax - b\|_2^2 + \lambda \|x\|_1 = g(x) + h(x)$$

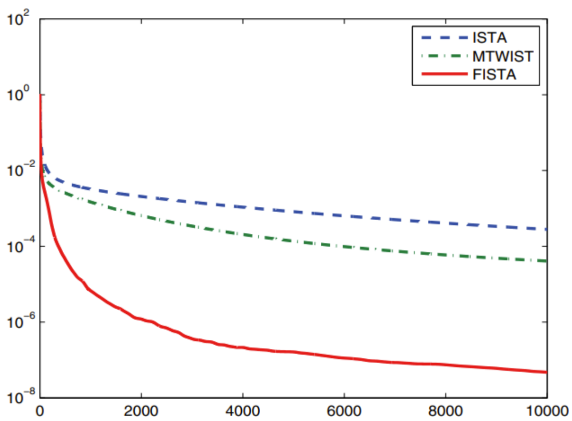
$$\Rightarrow \quad \begin{aligned} x_{k+1} &= SoftTheshold\left(y_k - t_k \nabla g(y_k), \lambda t_k\right) = SoftTheshold\left(y_k - t_k A^T (Ay_k - b), \lambda t_k\right) \\ y_{k+1} &= x_{k+1} + \frac{k}{k+3}(x_{k+1} - x_k) \end{aligned}$$

FISTA -- Accelerating ISTA

SIAM Publications Library
https://epubs.siam.org › doi

A Fast Iterative Shrinkage-Thresholding Algorithm for ...

by A Beck · 2009 · Cited by 12694 — We consider the class of iterative shrinkage-thresholding algorithms (ISTA) for solving **linear inverse problems** arising in signal/image processing.



$F(\mathbf{x}_k) - F(\mathbf{x}^*)$ *of ISTA, MTWIST, and FISTA.*

Enhancing Recovery Accuracy

L_p-norm (p < 1) Minimization

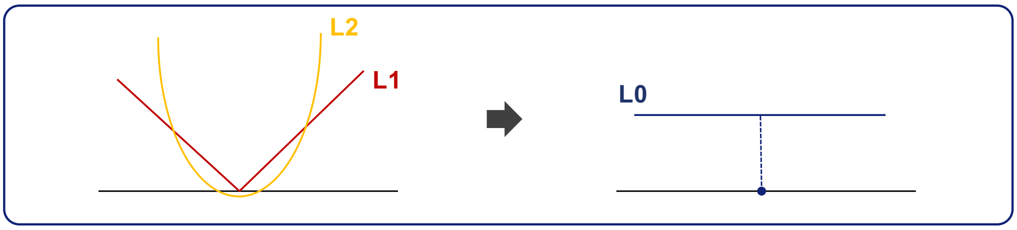
Revisiting L_k-norm, but k<1

- L_k-norm

Definition:	$\|\|x\|\|_k = (\|x_1\|^k + \|x_2\|^k + \|x_3\|^k + ... + \|x_n\|^k)^{\frac{1}{k}}$
L2-norm:	$\|\|x\|\|_2 = \sqrt{\|x_1\|^2 + \|x_2\|^2 + \|x_3\|^2 + ... + \|x_n\|^2}$
L1-norm:	$\|\|x\|\|_1 = \|x_1\| + \|x_2\| + \|x_3\| + ... + \|x_n\|$

- Theoretically, however, the ideal solver for sparse signal processing is **L0-norm** (counting # of non-zero elements)

L0-norm: $\|x\|_0 = \sum_j 1 \{x_j \neq 0\}$



➔ Minimizing L0-norm is a **non-convex** optimization (**Hard to implement** in practice)

Proximity Operator for L_0 Minimization

• Proximity (Thesholding) operator for L_0-norm

$$\underset{x}{\text{minimize}} \; f(x) = \|Ax-b\|_2^2 + \lambda\|x\|_0 = g(x) + h(x)$$

$$\Rightarrow \quad x_{k+1} = \underset{x}{\text{argmin}}\left[g(x_k) + \frac{1}{2t_k}\|x-(x_k - t_k\nabla g(x_k))\|_2^2 + h(x)\right]$$

$$= \underset{x}{\text{argmin}}\left[\frac{1}{2t_k}\|x-(x_k - t_k\nabla g(x_k))\|_2^2 + \lambda\|x\|_0\right]$$

$$= \underset{x}{\text{argmin}}\left[\frac{1}{2\lambda t_k}\|x-(x_k - t_k\nabla g(x_k))\|_2^2 + \|x\|_0\right]$$

$$= HardThreshold\left(x_k - t_k\nabla g(x_k), \lambda t_k\right) = HardThreshold\left(x_k - t_k A^T(Ax_k - b), \sqrt{2\lambda t_k}\right)$$

$$= \begin{cases} x_k - t_k A^T(Ax_k - b) & (|x_k - t_k A^T(Ax_k - b)| \geq \sqrt{2\lambda t_k}) \\ 0 & (|x_k - t_k A^T(Ax_k - b)| < \sqrt{2\lambda t_k}) \end{cases}$$

Approximation to L_0-norm -- Reweighted L_1-norm

• Approximated L_0-norm by **reweighted L1-norm**

L0-norm: $\underset{x}{\text{minimize}} \; \|x\|_0 = \sum_j 1\ \{x_j \neq 0\}$

L1-norm with W: $\underset{x}{\text{minimize}} \; \|Wx\|_1 = \sum_j w_j \cdot |x_j| \qquad where \;\; w_j = \frac{1}{|x_j| + \delta} \quad (\delta > 0)$

$$\underset{x}{\text{minimize}} \; \|Wx\|_1 = \sum_j \frac{|x_j|}{|x_j| + \delta} \approx \sum_j 1\ \{x_j \neq 0\}$$

S. Boyd, "Enhancin Sparsity by Reweighted l_1 Minimization", J Fourier Anal Appl (2008) 14:877-905

Graphical Interpretation of Reweighted L_1-norm

- Approximated L_0-norm by **reweighted L1-norm**

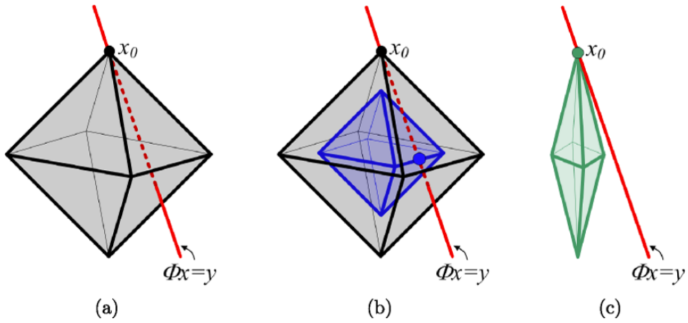


Fig. 1 Weighting ℓ_1 minimization to improve sparse signal recovery. **(a)** Sparse signal x_0, feasible set $\Phi x = y$, and ℓ_1 ball of radius $\|x_0\|_{\ell_1}$. **(b)** There exists an $x \neq x_0$ for which $\|x\|_{\ell_1} < \|x_0\|_{\ell_1}$. **(c)** Weighted ℓ_1 ball. There exists no $x \neq x_0$ for which $\|Wx\|_{\ell_1} \leq \|Wx_0\|_{\ell_1}$

S. Boyd, "Enhancin Sparsity by Reweighted l_1 Minimization", J Fourier Anal Appl (2008) 14:877-905

Implementation of Reweighted L_1-norm

- Pseudo-code of **reweighted L_1-norm**

Step 1. Implement conventional L_1-norm
w_j=1

$$\underset{x}{\text{minimize}} \quad \|Ax-b\|_2^2 + \lambda\|x\|_1$$
$$= \|Ax-b\|_2^2 + \lambda \sum_j |x_j|$$

Step 2. With resulting x from step 1 ($x^{(1)}$),
compute reweighting matrix W

$$w_j^{(1)} = \frac{1}{|x_j^{(1)}| + \delta} \quad (\delta > 0)$$

Step 3. Execute L_1 Minimization with W
(initial guess : $x^{(1)}$)

$$\underset{x}{\text{minimize}} \quad \|Ax-b\|_2^2 + \lambda \|W^{(1)}x\|_1$$
$$= \|Ax-b\|_2^2 + \lambda \sum_j w_j^{(1)} \cdot |x_j|$$

Step 4. Repeat [Step 2] to [Step 3]
until convergence

$$w_j^{(k)} = \frac{1}{|x_j^{(k)}| + \delta} \quad (\delta > 0)$$

$$\underset{x}{\text{minimize}} \quad \|Ax-b\|_2^2 + \lambda \|W^{(k)}x\|_1$$

L_p-norm ($p < 1$) Minimization for Total-Variation

YONSEI UNIVERSITY YCC PHYSICS

Implementation of Total-Variation Minimization by Reweighted L_1-norm

- Pseudo-code of **Total-Variation reweighted L_1-norm**

Step 1. Implement conventional L_1-norm

$$\underset{x}{\text{minimize}} \quad \frac{\mu}{2}\|Ax-b\|_2^2 + \|D_h x\|_1 + \|D_v x\|_1$$

Step 2. With resulting x from step 1 ($x^{(1)}$), compute reweighting matrix W

$$w_{h,j}^{(1)} = \frac{1}{|D_h x_j^{(1)}| + \delta} \ (\delta > 0), \quad w_{v,j}^{(1)} = \frac{1}{|D_v x_j^{(1)}| + \delta} \ (\delta > 0)$$

Step 3. Execute L_1 Minimization with W (initial guess : $x^{(1)}$)

$$\underset{x}{\text{minimize}} \quad \frac{\mu}{2}\|Ax-b\|_2^2 + \left\|W_h^{(1)}(D_h x)\right\|_1 + \left\|W_v^{(1)}(D_v x)\right\|_1$$

Step 4. Repeat [Step 2] to [Step 3] until convergence

$$w_{h,j}^{(k)} = \frac{1}{|D_h x_j^{(k)}| + \delta} \ (\delta > 0), \quad w_{v,j}^{(k)} = \frac{1}{|D_v x_j^{(k)}| + \delta} \ (\delta > 0)$$

$$\underset{x}{\text{minimize}} \quad \frac{\mu}{2}\|Ax-b\|_2^2 + \left\|W_h^{(k)}(D_h x)\right\|_1 + \left\|W_v^{(k)}(D_v x)\right\|_1$$

L_0-norm with ADMM -- Reweighting Process

- Denoising with TV min.

$$\underset{x}{\text{minimize}} \; f(x) = g(x) + h_h(x) + h_v(x) = \frac{\mu}{2}\|x-f\|_2^2 + \boxed{\|W_h(D_h x)\|_1 + \|W_v(D_v x)\|_1}$$

$$\Rightarrow \underset{x,d_h,d_v,b_h,b_v}{\text{minimize}} \; \frac{\mu}{2}\|x-f\|_2^2 + |d_h| + |d_v| + \frac{\lambda}{2}\|d_h - W_h(D_h x) - q_h\|_2^2 + \frac{\lambda}{2}\|d_v - W_v(D_v x) - q_v\|_2^2$$

Step 1. Update x: $\underset{x}{\text{minimize}} \; \frac{\mu}{2}\|x-f\|_2^2 + \frac{\lambda}{2}\|d_h - W_h(D_h x) - q_h\|_2^2 + \frac{\lambda}{2}\|d_v - W_v(D_v x) - q_v\|_2^2$

Step 2. Update d_u, d_v:

$$\underset{d_h}{\text{minimize}} \; |d_h| + \frac{\lambda}{2}\|d_h - W_h(D_h x) - q_h\|_2^2 \quad \Rightarrow \quad d_h^{\,k+1} = SoftThreshold\left(W_h(D_h x^{k+1}) + q_h^{\,k}, \frac{1}{\lambda}\right)$$

Step 3. Update q_h, q_v:

$$\underset{q_h}{\text{minimize}} \; \frac{\lambda}{2}\|d_h - W_h(D_h x) - q_h\|_2^2 \quad \Rightarrow \quad q_h^{\,k+1} = q_h^{\,k} + \left(W_h(D_h x^{k+1}) - d_h^{\,k+1}\right)$$

L_0-norm with Chambolle-Pock -- Reweighting Process

- Modifying the primal-dual solution (Chambolle-Pock algorithm)

$$F(Kx) = f_1(Ax) + f_2(D_h x) + f_3(D_v x), \; G(x) = [x]_+$$
$$f_1(y_1) = \frac{\lambda}{2}\|y_1 - b\|_2^2 \; (y_1 = Ax),$$
$$\boxed{\begin{aligned} f_2(y_2) &= \|W_h \cdot y_2\|_1 \; (y_2 = D_h x), \\ f_3(y_3) &= \|W_v \cdot y_3\|_1 \; (y_3 = D_v x) \end{aligned}}$$

$$\begin{aligned} y_l^{\,n+1} &= prox_{\sigma F^*}(y_l^{\,n} + \sigma \cdot K\tilde{x}^n) \\ x^{n+1} &= prox_{\tau G}(x^n - \tau \textstyle\sum_l K^T y_l^{\,n+1}) \\ \tilde{x}^{n+1} &= x^{n+1} + \theta(x^{n+1} - x^n) \end{aligned}$$

$$\begin{aligned} y_2^{\,n+1} &= prox_{\sigma f_2^*}(y_2^{\,n} + \sigma \cdot W_h \cdot (D_h \tilde{x}^n)) = Truncate(y_2^{\,n} + \sigma \cdot W_h \cdot (D_h \tilde{x}^n), \sigma) \\ &= \begin{cases} y_2^{\,n} + \sigma \cdot W_h \cdot (D_h \tilde{x}^n) & (|\, y_2^{\,n} + \sigma \cdot W_h \cdot (D_h \tilde{x}^n)| < \sigma) \\ \sigma & (|\, y_2^{\,n} + \sigma \cdot W_h \cdot (D_h \tilde{x}^n)| < \sigma) \end{cases} \end{aligned}$$

$$\begin{aligned} x^{n+1} &= prox_{\tau G}(x^n - \tau \textstyle\sum_l K^T y_l^{\,n+1}) \\ &= \left[x^n - \tau(A^T y_1^{\,n+1} + W_h(D_h^{\,T} y_2^{\,n+1}) + W_v(D_v^{\,T} y_3^{\,n+1}))\right]_+ \end{aligned}$$

L_0-norm with Chambolle-Pock

- Modifying the primal-dual solution (Chambolle-Pock algorithm)

$$\underset{x}{\text{minimize}} \quad F(Kx)+G(x)$$

$$\begin{aligned} y_l^{n+1} &= prox_{\sigma F^*}(y_l^n+\sigma\cdot K\tilde{x}^n) \\ x^{n+1} &= prox_{\tau G}(x^n-\tau\sum_l K^T y_l^{n+1}) \\ \tilde{x}^{n+1} &= x^{n+1}+\theta(x^{n+1}-x^n) \end{aligned} \quad\Rightarrow\quad \begin{aligned} y^{n+1} &= prox_{\sigma^{-1}F}(\sigma^{-1}y^n+Kx^n) \\ z^{n+1} &= z^n+\sigma(Kx^n-y^{n+1}) \\ x^{n+1} &= prox_{\tau G}(x^n-\tau\sum_l K^T y_l^{n+1}) \\ x^{n+1} &= x^{n+1}+\theta(x^{n+1}-x^n) \end{aligned}$$

*Moreau's decomposition

$$x = prox_{\sigma F}(x)+\sigma\cdot prox_{\sigma^{-1}F^*}(\sigma^{-1}x)$$

$$\Rightarrow \quad prox_{\sigma F^*}(\tilde{y}) = \tilde{y}-\sigma\cdot prox_{\sigma^{-1}F}(\sigma^{-1}\tilde{y})$$

T. Mollenhoff et al, Low Rank Priors for Color Image Regularization. EMMCVPR 2015

Example of L_0 Minimization by Chambolle-Pock

$$\underset{x}{\text{minimize}} \quad F(Kx)+G(x)=\frac{\mu}{2}\|Ax-b\|_2^2+\|x\|_0 \ (x\ge 0)$$

$$F(Kx)=F\left(\begin{bmatrix} A \\ I \end{bmatrix}x\right)=f_1(Ax)+f_2(x), \ G(x)=[x]_+$$

$$f_1(y_1)=\frac{\mu}{2}\|y_1-b\|_2^2 \ (y_1=Ax), \qquad f_2(y_2)=\|y_2\|_1 \ (y_2=x)$$

$$\begin{aligned} z^{n+1} &= prox_{\sigma^{-1}F}(\sigma^{-1}y^n+Kx^n) \\ y^{n+1} &= y^n+\sigma(Kx^n-z^{n+1}) \\ x^{n+1} &= prox_{\tau G}(x^n-\tau\sum_l K^T y_l^{n+1}) \\ x^{n+1} &= x^{n+1}+\theta(x^{n+1}-x^n) \end{aligned}$$

Step 1. Update z's:

$$z_1^{n+1} = prox_{\sigma^{-1}F}(\sigma^{-1}y_1^n+A\tilde{x}^n)=\frac{(\sigma^{-1}y_1^n+A\tilde{x}^n)+\mu b/\sigma}{1+(\mu/\sigma)}$$

$$z_2^{n+1} = prox_{\sigma^{-1}F}(\sigma^{-1}y_2^n+\tilde{x}^n)=\begin{cases} \sigma^{-1}y_2^n+\tilde{x}^n & (|\sigma^{-1}y_2^n+\tilde{x}^n|\ge\sigma^{-1}) \\ 0 & (|\sigma^{-1}y_2^n+\tilde{x}^n|<\sigma^{-1}) \end{cases}$$

Step 2. Update y's:

$$y_1^{n+1}=y_1^n+\sigma(Ax^n-z_1^{n+1})$$

$$y_2^{n+1}=y_2^n+\sigma(x^n-z_2^{n+1})$$

Step 3. Update x and $\tilde{x}$:

$$x^{n+1}=prox_{\tau G}(x^n-\tau\sum_l K^T y_l^{n+1})=\left[x^n-\tau(A^T y_1^{n+1}+y_2^{n+1})\right]_+$$

$$\tilde{x}^{n+1}=x^{n+1}+\theta(x^{n+1}-x^n)$$

Example of Total-Variation L_0 Minimization by Chambolle-Pock

$$F(Kx) = F\left(\begin{bmatrix} A \\ D_h \\ D_v \end{bmatrix} x\right) = f_1(Ax) + f_2(D_h x) + f_3(D_v x), \quad G(x) = [x]_+$$

$$f_1(y_1) = \frac{\mu}{2}\|y_1 - b\|_2^2 \ (y_1 = Ax),$$

$$f_2(y_2) = \|y_2\|_1 \,(y_2 = D_h x), \quad f_3(y_3) = \|y_3\|_1 \,(y_3 = D_v x)$$

$$z^{n+1} = prox_{\sigma^{-1}F}(\sigma^{-1} y^n + Kx^n)$$
$$y^{n+1} = y^n + \sigma(Kx^n - z^{n+1})$$
$$x^{n+1} = prox_{\tau G}(x^n - \tau \sum_l K^T y_l^{n+1})$$
$$x^{n+1} = x^{n+1} + \theta(x^{n+1} - x^n)$$

Step 1. Update z's:

$$z_1^{n+1} = prox_{\sigma^{-1}F}(\sigma^{-1} y_1^n + A\tilde{x}^n) = \frac{(\sigma^{-1} y_1^n + A\tilde{x}^n) + \mu b/\sigma}{1 + (\mu/\sigma)}$$

$$z_2^{n+1} = prox_{\sigma^{-1}F}(\sigma^{-1} y_2^n + D_h\tilde{x}^n) = \begin{cases} \sigma^{-1} y_2^n + D_h\tilde{x}^n & (|\sigma^{-1} y_2^n + D_h\tilde{x}^n| \geq \sigma^{-1}) \\ 0 & (|\sigma^{-1} y_2^n + D_h\tilde{x}^n| < \sigma^{-1}) \end{cases}$$

$$z_3^{n+1} = prox_{\sigma^{-1}f_3}(\sigma^{-1} y_3^n + D_v\tilde{x}^n) = \begin{cases} \sigma^{-1} y_3^n + D_v\tilde{x}^n & (|\sigma^{-1} y_3^n + D_v\tilde{x}^n| \geq \sigma^{-1}) \\ 0 & (|\sigma^{-1} y_3^n + D_v\tilde{x}^n| < \sigma^{-1}) \end{cases}$$

YONSEI UNIVERSITY YCC PHYSICS

Example of Total-Variation L_0 Minimization by Chambolle-Pock

$$F(Kx) = F\left(\begin{bmatrix} A \\ D_h \\ D_v \end{bmatrix} x\right) = f_1(Ax) + f_2(D_h x) + f_3(D_v x), \quad G(x) = [x]_+$$

$$f_1(y_1) = \frac{\mu}{2}\|y_1 - b\|_2^2 \ (y_1 = Ax),$$

$$f_2(y_2) = \|y_2\|_1 \,(y_2 = D_h x), \quad f_3(y_3) = \|y_3\|_1 \,(y_3 = D_v x)$$

$$z^{n+1} = prox_{\sigma^{-1}F}(\sigma^{-1} y^n + Kx^n)$$
$$y^{n+1} = y^n + \sigma(Kx^n - z^{n+1})$$
$$x^{n+1} = prox_{\tau G}(x^n - \tau \sum_l K^T y_l^{n+1})$$
$$x^{n+1} = x^{n+1} + \theta(x^{n+1} - x^n)$$

Step 2. Update y's:

$$y_1^{n+1} = y_1^n + \sigma(Ax^n - z_1^{n+1})$$
$$y_2^{n+1} = y_2^n + \sigma(D_h x^n - z_2^{n+1})$$
$$y_3^{n+1} = y_3^n + \sigma(D_v x^n - z_3^{n+1})$$

Step 3. Update x and $\tilde{x}$:

$$x^{n+1} = prox_{\tau G}(x^n - \tau \sum_l K^T y_l^{n+1}) = \left[x^n - \tau(A^T y_1^{n+1} + y_2^{n+1} + y_3^{n+1})\right]_+$$

$$\tilde{x}^{n+1} = x^{n+1} + \theta(x^{n+1} - x^n)$$

YONSEI UNIVERSITY
COLLEGE OF MEDICINE