1. Generalized nuclear norm for patches

Reference: [2012 Chartrand] Nonconvex splitting for regularized low-rank sparse decomposition

Consider the minimization problem involving nuclear norm:

$$\mathbf{X} = \arg\min f(\mathbf{X}) + \sum_i H\big(\boldsymbol{\sigma}(\mathbf{P}_i\mathbf{X})\big)$$

where $\mathbf{P}_i$ is the patch extraction matrix for the ith patch.

1.1.Selection of *H*

Let H be elementwise generalized Huber function

$$H(\mathbf{X}) \coloneqq H_{\mu,p}(\mathbf{X}) \coloneqq \sum h_{\mu,p}(x)$$

where

$$h_{\mu,p}(x) = \begin{cases} \dfrac{|x|^2}{2\mu} & |x| \le \mu^{\frac{1}{2-p}} \\ \dfrac{|x|^p}{p} - \delta & |x| \ge \mu^{\frac{1}{2-p}} \end{cases}$$

where $\delta = \left(\frac{1}{p} - \frac{1}{2}\right)\mu^{\frac{1}{2-p}}$ to make the function C^1 continuous.

Define an auxiliary function g as:

$$\frac{|y|^2}{2} + \mu g_{\mu,p}(y) = \left(\frac{|\cdot|^2}{2} - \mu h_{\mu,p}\right)^*(y)$$

where the conjugate is given by Legendre-Frenchel transform

$$f^*(y) = \max_x\{xy - f(x)\}$$

$g_{\mu,p}$ has several properties:

(1) Proximal mapping

$$h_{\mu,p}(x) = \min_y g_{\mu,p}(y) + \frac{1}{2\mu}(x-y)^2$$

(2) Soft thresholding: the minimizer y^* for problem (1) is given by

$$y^* = shrink_p(x,\mu) \coloneqq \max\{0, |x| - \mu|x|^{p-1}\}\frac{x}{|x|}$$

(3) Generalize to nuclear norm

Let

$$G_{\mu,p}(\mathbf{Y}) = \sum g_{\mu,p}(y)$$

Then

$$\min_{\mathbf{\Theta}} G_{\mu,p}(\mathbf{\Theta}) + \frac{1}{2\mu}\|\mathbf{\Theta} - \mathbf{\Sigma}\|_F^2 = \min_{\mathbf{Y}} G_{\mu,p}\big(\boldsymbol{\sigma}(\mathbf{Y})\big) + \frac{1}{2\mu}\|\mathbf{Y} - \mathbf{X}\|_F^2$$

where $\mathbf{\Theta}$, $\mathbf{\Sigma}$ are the singular value matrices of $\mathbf{Y}$ and $\mathbf{X}$.

The minimizer $\mathbf{Y}^*$ is given by

$$\mathbf{Y}^* = \mathbf{U}shrink_p(\mathbf{\Sigma}, \mu)\mathbf{V}^*$$

where $\mathbf{X} = \mathbf{U\Sigma V}^*$

1.2.Building Cost function

$$H_{\mu,p}\big(\boldsymbol{\sigma}(\mathbf{P}_i\mathbf{X})\big) = \min_{\mathbf{\Theta}} G_{\mu,p}(\mathbf{\Theta}) + \frac{1}{2\mu}\|\mathbf{\Theta} - \boldsymbol{\sigma}(\mathbf{P}_i\mathbf{X})\|_F^2 = \min_{\mathbf{Y}} G_{\mu,p}\big(\boldsymbol{\sigma}(\mathbf{Y})\big) + \frac{1}{2\mu}\|\mathbf{Y} - \mathbf{P}_i\mathbf{X}\|_F^2$$

Then the original problem is transformed into

$$\begin{aligned}
&\mathbf{X} = \arg\min f(\mathbf{X}) + \sum_i \min_{\mathbf{Y}_i} G_{\mu,p}\big(\boldsymbol{\sigma}(\mathbf{Y}_i)\big) + \frac{1}{2\mu}\|\mathbf{Y}_i - \mathbf{P}_i\mathbf{X}\|_F^2 \\
&= \arg\min_{\mathbf{X},\mathbf{Y}_i} f(\mathbf{X}) + \sum_i G_{\mu,p}\big(\boldsymbol{\sigma}(\mathbf{Y}_i)\big) + \frac{1}{2\mu}\|\mathbf{Y}_i - \mathbf{P}_i\mathbf{X}\|_F^2 \\
&\coloneqq \arg\min F(\mathbf{X}, \mathbf{Y})
\end{aligned}$$

Alternating minimization:

$$\mathbf{Y}_i^{k+1} = \arg\min_{\mathbf{Y}_i} G_{\mu,p}\big(\boldsymbol{\sigma}(\mathbf{Y}_i)\big) + \frac{1}{2\mu}\left\|\mathbf{Y}_i - \mathbf{P}_i\mathbf{X}^k\right\|_F^2 = \mathbf{U}_i^k shrink_p\big(\boldsymbol{\sigma}(\mathbf{P}_i X^k), \mu\big)\mathbf{V}_i^{k^*}$$

$$\mathbf{X}^{k+1} = \arg\min_{\mathbf{X}} S\big(\mathbf{X}; \mathbf{X}^k, \mathbf{Y}^{k+1}\big)$$

where $S\big(\mathbf{X}; \mathbf{X}^k, \mathbf{Y}^{k+1}\big)$ is the surrogate function for

$$f(\mathbf{X}) + \frac{1}{2\mu}\sum_i \left\|\mathbf{Y}_i^{k+1} - \mathbf{P}_i\mathbf{X}\right\|_F^2$$

The alternating minimization satisfies:

$$F\big(\mathbf{X}^{k+1}, \mathbf{Y}^{k+1}\big) \le F\big(\mathbf{X}^k, \mathbf{Y}^{k+1}\big) \le F\big(\mathbf{X}^k, \mathbf{Y}^k\big)$$

Giving a monotonic algorithm.

For spectral CT, the function became

$$\sum_s \left\{ f_s(\mathbf{x}_s) + \frac{1}{2\mu} \sum_i \left\| \mathbf{y}_{is}^{k+1} - \mathbf{P}_i \mathbf{x}_s \right\|_2^2 \right\}$$

which can be solved spectrum by spectrum.

2. SQS for TV norm

Reference: [2018] A separable quadratic surrogate total variation minimization algorithm for accelerating accurate CT reconstruction from few-views and limited-angle data

Consider the total variation penalty:

$$R(\mathbf{x}) = \sum_{i,j,k} \sqrt{\left(x_{i,j,k} - x_{i-1,j,k}\right)^2 + \left(x_{i,j,k} - x_{i,j-1,k}\right)^2 + \left(x_{i,j,k} - x_{i,j,k-1}\right)^2 + \epsilon^2} \coloneqq \sum_{i,j,k} R_{i,j,k}(\mathbf{x})$$

Define

$$h(y) \coloneqq \sqrt{y}; y_{i,j,k} = R_{i,j,k}^2(\mathbf{x})$$

So

$$h(y_{i,j,k}) = R_{i,j,k}(\mathbf{x})$$

Let

$$S_h(y; y^n) = h(y^n) + \frac{1}{2\sqrt{y^n}}(y - y^n)$$

We have Prop. 1:

$$S_h(y; y^n) = h(y^n)$$

$$S_h(y; y^n) \geq h(y)$$

Second equation is because

$$S_h(y; y^n) - h(y) = \sqrt{y^n} + \frac{1}{2\sqrt{y^n}}(y - y^n) - \sqrt{y}$$

$$= \frac{1}{2\sqrt{y^n}}\left(y - y^n + 2y^n - 2\sqrt{yy^n}\right) = \frac{1}{2\sqrt{y^n}}\left(\sqrt{y} - \sqrt{y^n}\right)^2 \geq 0$$

Let $y = y_{i,j,k}, y^n = y_{i,j,k}^n$:

$$S_{i,j,k}^1(\mathbf{x}; \mathbf{x}^n) \coloneqq S_h\left(y_{i,j,k}; y_{i,j,k}^n\right) = \frac{y_{i,j,k}}{2\sqrt{y_{i,j,k}^n}} + \frac{\sqrt{y_{i,j,k}^n}}{2}$$

$$= \frac{R_{i,j,k}(\mathbf{x}^n)}{2} + \frac{1}{2R_{i,j,k}(\mathbf{x}^n)}\left[\left(x_{i,j,k} - x_{i-1,j,k}\right)^2 + \left(x_{i,j,k} - x_{i,j-1,k}\right)^2 + \left(x_{i,j,k} - x_{i,j,k-1}\right)^2 + \epsilon^2\right]$$

According to Prop. 1, $S^1_{i,j,k}(\mathbf{x};\mathbf{x}^n)$ is a valid surrogate of $R_{i,j,k}(\mathbf{x})$ at $\mathbf{x}^n$.

Making further surrogate with convex relaxation:

$$\begin{aligned} x_{i,j,k} - x_{i-1,j,k} &= \frac{1}{2}\left[\left(2x_{i,j,k} - x^n_{i,j,k} - x^n_{i-1,j,k}\right) - \left(2x_{i-1,j,k} - x^n_{i-1,j,k} - x^n_{i,j,k}\right)\right] \\ &= \frac{1}{2}\left(2x_{i,j,k} - x^n_{i,j,k} - x^n_{i-1,j,k}\right) + \frac{1}{2}\left(-2x_{i-1,j,k} + x^n_{i-1,j,k} + x^n_{i,j,k}\right) \end{aligned}$$

Then we have

$$\left(x_{i,j,k} - x_{i-1,j,k}\right)^2 \leq \frac{1}{2}\left(2x_{i,j,k} - x^n_{i,j,k} - x^n_{i-1,j,k}\right)^2 + \frac{1}{2}\left(2x_{i-1,j,k} - x^n_{i-1,j,k} - x^n_{i,j,k}\right)^2$$

And the equality holds when $\mathbf{x} = \mathbf{x}^n$.

Do the same for the rest 2 dimensions and substitute into $S_{i,j,k}(\mathbf{x};\mathbf{x}^n)$ and

$$\begin{aligned} S^1_{i,j,k}(\mathbf{x};\mathbf{x}^n) \leq \frac{R_{i,j,k}(\mathbf{x}^n)}{2} + \frac{1}{2R_{i,j,k}(\mathbf{x}^n)}\Bigg[&\frac{1}{2}\left(2x_{i,j,k} - x^n_{i,j,k} - x^n_{i-1,j,k}\right)^2 + \frac{1}{2}\left(2x_{i-1,j,k} - x^n_{i-1,j,k} - x^n_{i,j,k}\right)^2 \\ &+ \frac{1}{2}\left(2x_{i,j,k} - x^n_{i,j,k} - x^n_{i,j-1,k}\right)^2 + \frac{1}{2}\left(2x_{i,j-1,k} - x^n_{i,j-1,k} - x^n_{i,j,k}\right)^2 \\ &+ \frac{1}{2}\left(2x_{i,j,k} - x^n_{i,j,k} - x^n_{i,j,k-1}\right)^2 + \frac{1}{2}\left(2x_{i,j,k-1} - x^n_{i,j,k-1} - x^n_{i,j,k}\right)^2 + \epsilon^2\Bigg] \\ \coloneqq S_{i,j,k}(\mathbf{x};\mathbf{x}^n) \end{aligned}$$

which is the final surrogate of $R_{i,j,k}(\mathbf{x})$. It is straightforward to verify that $S_{i,j,k}(\mathbf{x}^n;\mathbf{x}^n) = S^1_{i,j,k}(\mathbf{x}^n;\mathbf{x}^n)$.

The total surrogate is

$$S(\mathbf{x};\mathbf{x}^n) = \sum_{i,j,k} S_{i,j,k}(\mathbf{x};\mathbf{x}^n)$$

First order derivative:

$$\begin{aligned} &\frac{\partial S}{\partial x_{i,j,k}} \\ &= \frac{\partial S_{i,j,k}}{\partial x_{i,j,k}} + \frac{\partial S_{i+1,j,k}}{\partial x_{i,j,k}} + \frac{\partial S_{i,j+1,k}}{\partial x_{i,j,k}} + \frac{\partial S_{i,j,k+1}}{\partial x_{i,j,k}} \\ &= \frac{\left(2x_{i,j,k} - x^n_{i,j,k} - x^n_{i-1,j,k}\right) + \left(2x_{i,j,k} - x^n_{i,j,k} - x^n_{i,j-1,k}\right) + \left(2x_{i,j,k} - x^n_{i,j,k} - x^n_{i,j,k-1}\right)}{2R_{i,j,k}(\mathbf{x}^n)} \\ &+ \frac{2x_{i,j,k} - x^n_{i,j,k} - x^n_{i+1,j,k}}{2R_{i+1,j,k}(\mathbf{x}^n)} + \frac{2x_{i,j,k} - x^n_{i,j,k} - x^n_{i,j+1,k}}{2R_{i,j+1,k}(\mathbf{x}^n)} + \frac{2x_{i,j,k} - x^n_{i,j,k} - x^n_{i,j,k+1}}{2R_{i,j,k+1}(\mathbf{x}^n)} \end{aligned}$$

And

$$\left.\frac{\partial S}{\partial x_{i,j,k}}\right|_{\mathbf{x}=\mathbf{x}^n} = \frac{\left(x^n_{i,j,k} - x^n_{i-1,j,k}\right) + \left(x^n_{i,j,k} - x^n_{i,j-1,k}\right) + \left(x^n_{i,j,k} - x^n_{i,j,k-1}\right)}{2R_{i,j,k}(\mathbf{x}^n)}$$

$$+\frac{x_{i,j,k}^n - x_{i+1,j,k}^n}{2R_{i+1,j,k}(\mathbf{x}^n)} + \frac{x_{i,j,k}^n - x_{i,j+1,k}^n}{2R_{i,j+1,k}(\mathbf{x}^n)} + \frac{x_{i,j,k}^n - x_{i,j,k+1}^n}{2R_{i,j,k+1}(\mathbf{x}^n)}$$

Second order derivative:

$$\frac{\partial^2 S}{\partial x_{i,j,k}^2} = \frac{6}{2R_{i,j,k}(\mathbf{x}^n)} + \frac{2}{2R_{i+1,j,k}(\mathbf{x}^n)} + \frac{2}{2R_{i,j+1,k}(\mathbf{x}^n)} + \frac{2}{2R_{i,j,k+1}(\mathbf{x}^n)}$$
$$= \frac{3}{R_{i,j,k}(\mathbf{x}^n)} + \frac{1}{R_{i+1,j,k}(\mathbf{x}^n)} + \frac{1}{R_{i,j+1,k}(\mathbf{x}^n)} + \frac{1}{R_{i,j,k+1}(\mathbf{x}^n)}$$

For implementation:

First calculate $R_{i,j,k}(\mathbf{x}^n)$ for each pixel and store the array;

Then calculate first and second order derivatives for each pixel.

3. Optimization Algorithm for TV + nuclear norm

The total cost function for spectral CT with be

$$\mathbf{X} = \arg\min \sum_s \left\{\frac{1}{2}\|\mathbf{A}\mathbf{x}_s - \mathbf{b}_s\|_{\mathbf{w}_s}^2 + \beta_s \|\mathbf{x}_s\|_{TV}\right\} + \beta \sum_i H_{\mu,p}\big(\boldsymbol{\sigma}(\mathbf{X}_i)\big)$$

where

$$\mathbf{X}_i = [\mathbf{P}_i\mathbf{x}_1, \mathbf{P}_i\mathbf{x}_2, \dots, \mathbf{P}_i\mathbf{x}_S]$$

is a matrix.

Minimization w.r.t. nuclear norm:

$$\mathbf{X}_i^n = \mathbf{U}_i^n \boldsymbol{\sigma}(\mathbf{X}_i) \mathbf{V}_i^{n*}$$

$$\boldsymbol{\sigma}\big(\mathbf{Y}_i^{n+1}\big) = shrink_p(\boldsymbol{\sigma}(\mathbf{X}_i^n), \mu) = \max\{0, |\boldsymbol{\sigma}(\mathbf{X}_i^n)| - \mu|\boldsymbol{\sigma}(\mathbf{X}_i^n)|^{p-1}\} \frac{\boldsymbol{\sigma}(\mathbf{X}_i^n)}{\big|\boldsymbol{\sigma}\big(\mathbf{X}_i^n\big)\big|}$$

$$\mathbf{Y}_i^{n+1} = \mathbf{U}_i^n \boldsymbol{\sigma}\big(\mathbf{Y}_i^{n+1}\big) \mathbf{V}_i^{n*}$$

Then minimizing w.r.t. surrogate of TV and patching for each spectrum independently:

$$\mathbf{x}_s^{n+1} = \mathbf{x}_s^n - \frac{\mathbf{A}^T \mathbf{w}_s (\mathbf{A}\mathbf{x}_s - \mathbf{b}_s) + \frac{\beta}{2\mu} \sum_i \mathbf{P}_i^T (\mathbf{P}_i \mathbf{x}_s^n - \mathbf{y}_s^{n+1}) + \beta_s \frac{\partial S(\mathbf{x}_s^n; \mathbf{x}_s^n)}{\partial \mathbf{x}}}{\mathbf{A}^T \mathbf{w}_s \mathbf{A} \mathbf{1} + \frac{\beta}{2\mu} \sum_i \mathbf{P}_i^T \mathbf{P}_i \mathbf{1} + \beta_s \frac{\partial^2 S(\mathbf{x}_s^n; \mathbf{x}_s^n)}{\partial \mathbf{x}^2}}$$

The TV related terms are defined above.

4. SQS for Non-local weighted Gaussian prior

Prior function is defined as:

$$R(\mathbf{x}) = \sum_{N}\sum_{k} w_{jk}\left(x_j - x_k\right)^2$$

Use convex splitting:

$$\left(x_j - x_k\right)^2 = \left[\frac{1}{2}\left(2x_j - x_j^{(n)} - x_k^{(n)}\right) + \frac{1}{2}\left(-2x_k + x_k^{(n)} + x_j^{(n)}\right)\right]^2$$
$$\leq \frac{1}{2}\left(2x_j - x_j^{(n)} - x_k^{(n)}\right)^2 + \frac{1}{2}\left(2x_k - x_k^{(n)} - x_j^{(n)}\right)^2$$

Equality holds at $\mathbf{x} = \mathbf{x}^{(n)}$.

Surrogate:

$$Q(\mathbf{x}) = \sum_{N}\sum_{k} w_{jk}\left[\frac{1}{2}\left(2x_j - x_j^{(n)} - x_k^{(n)}\right)^2 + \frac{1}{2}\left(2x_k - x_k^{(n)} - x_j^{(n)}\right)^2\right]$$

First order derivative:

$$\frac{\partial Q}{\partial x_j} = 2\sum_{k} w_{jk}\left(2x_j - x_j^{(n)} - x_k^{(n)}\right) + w_{kj}\left(2x_j - x_j^{(n)} - x_k^{(n)}\right)$$

Second order derivative:

$$\frac{\partial^2 Q}{\partial x_j^2} = 4\sum_{k} w_{jk} + w_{kj}$$

Derivatives at current position:

$$\left.\frac{\partial Q}{\partial x_j}\right|_{x_j^{(n)}} = 2\sum_{k} w_{jk}\left(x_j^{(n)} - x_k^{(n)}\right) + w_{kj}\left(x_j^{(n)} - x_k^{(n)}\right)$$

$$\left.\frac{\partial^2 Q}{\partial x_j^2}\right|_{x_j^{(n)}} = 4\sum_{k} w_{jk} + w_{kj}$$

For most cases, we have

$$w_{kj} = w_{jk}$$

$$\sum_{k} w_{jk} = 1$$

And the derivatives became:

$$\left.\frac{\partial Q}{\partial x_j}\right|_{x_j^{(n)}} = 4\sum_{k} w_{jk}\left(x_j^{(n)} - x_k^{(n)}\right) = 4x_j^{(n)} - 4\sum_{k} w_{jk}x_k^{(n)} = 4\left[\mathbf{x}^{(n)} - Guided\_filter\left(\mathbf{x}^{(n)}\right)\right]$$

$$\left.\frac{\partial^2 Q}{\partial x_j^2}\right|_{x_j^{(n)}} = 8$$

The loss can be calculated as

$$R(\mathbf{x}) = \sum_N \sum_k w_{jk}\left(x_j - x_k\right)^2 = \sum_n \left(x_j^2 \sum_k w_{jk} - 2x_j \sum_k w_{jk} x_k + \sum_k w_{jk} x_k^2\right)$$

$$= \sum_n \left(x_j^2 - 2x_j Filter(x)_j + Filter(x^2)_j\right)$$