

Algorithmes optimisés de minimisation de la variation totale.

DOCUMENTATION.

Fabien PIERRE.

RÉSUMÉ.

Le document suivant donne des pistes pour comprendre comment fonctionnent les fonctions qui ont été implémentées dans cette boîte à outils Matlab. Pour l'utilisateur qui ne serait pas intéressé par une théorie roborative, l'exécution des cellules (Ctrl + Entrée) du fichier `script.m` lui donnera une explication directe.

Table des matières

1 Fonctions de base.	3
1.1 À propos de la transformation de Legendre-Fenchel.	3
1.2 Utilisation de la transformée de Legendre-Fenchel pour la sous-différentielle.	4
1.3 Application de la transformation de Legendre-Fenchel à la minimisation de la variation totale.	5
1.4 Application au débruitage d'image.	6
1.4.1 Considérations théoriques en vue de la construction de l'algorithme.	6
1.5 L'algorithme de Forward-Backward.	6
1.5.1 Cadre et présentation : le point de vue proximal.	6
1.5.2 Convergence.	7
1.5.3 Implémentation.	7
1.6 Application à la déconvolution.	7
1.7 Application à l'inpainting.	7
2 Utilisation d'algorithmes "primal-dual" pour des problèmes non lisses.	8
2.1 Le critère $TV-L^1$. Résolution par algorithme primal-dual.	8
2.1.1 Mise en place théorique de l'algorithme primal-dual.	8
2.2 Inpainting par minimisation de la variation totale : approche par algorithme primal-dual.	9

1 Fonctions de base.

Définition (La sous-différentielle.)

Soit Γ_0 l'ensemble des fonctions convexes de $\mathbb{R}^N \mapsto \mathbb{R} \cup \{+\infty\}$ semi-continue inférieurement, i.e.

$$\Gamma_0 := \{f \text{ tq } \forall x_0 \in \mathbb{R}^N, \forall \epsilon > 0, \exists V \text{ voisinage de } x_0, \forall x \in V : f(x) \leq f(x_0) - \epsilon\}.$$

On définit¹ la sous-différentielle au point x , l'ensemble

$$\partial f(x) := \{u \in \mathbb{R}^N \text{ tq } \forall y \in \mathbb{R}^N, \langle y - x | u \rangle + f(x) \leq f(y)\}.$$

Définition (Opérateur proximal.)

Soient $f \in \Gamma_0$, $x \in \mathbb{R}^N$ le problème suivant a une unique solution :

$$\min_{y \in \mathbb{R}^N} f(y) + \frac{1}{2} \|x - y\|^2,$$

notée $\text{prox}_f(x)$ appelé l'opérateur proximal de f .

Proposition 1

Soit $f \in \Gamma_0$. L'opérateur proximal de f est caractérisé par :

$$\forall (x, y) \in \mathbb{R}^N \times \mathbb{R}^N, y = \text{prox}_f(x) \Leftrightarrow x - y \in \partial f(y).$$

Remarque : 1 Dans ce cas particulier, l'opérateur proximal peut prendre la notation suivante :

$$z := \text{prox}_f(x) = (I_d + \partial f)^{-1}(x).$$

Posons pour cela z tel que

$$f(z) + \frac{1}{2} \|x - z\|^2 = \min_{y \in \mathbb{R}^N} f(y) + \frac{1}{2} \|x - y\|^2.$$

Alors on a $z := \text{prox}_f(x)$ et donc $x - z \in \partial f(z)$. Ou encore $x \in z + \partial f(z)$. Ce que l'on note

$$x = (I_d + \partial f(z)).$$

On note cela formellement :

$$z = (I_d + \partial f)^{-1}(x).$$

1.1 À propos de la transformation de Legendre-Fenchel.

Définition

On définit (dans [4] par exemple) la transformée de Legendre-Fenchel par $J^*(v) = \sup_u \langle u | v \rangle - J(u)$.

Proposition 2

Soit J une fonctionnelle homogène de degré 1 (i.e. $\forall u \in \mathbb{R}^N$ et $\forall \lambda > 0$ on a $J(\lambda u) = \lambda J(u)$.) Alors l'analyse convexe nous dit que

$$J^*(v) = 1_K(v).$$

Corollaire 1

Soit J une fonctionnelle homogène de degré 1, alors $J^{**}(u) = J(u)$.

Corollaire 2

Soit J une fonctionnelle homogène de degré 1, alors $J(v) = \sup_{u \in K} \langle u | v \rangle$.

1. [4]

Preuve En effet, par le corollaire 1 alors :

$$J(u) = J^{**}(u) = \sup_{v \in \mathbb{R}^N} \{ \langle u|v \rangle - J^*(u) \} ,$$

mais d'après la proposition 2 alors :

$$J(u) = J^{**}(u) = \sup_{v \in \mathbb{R}^N} \{ \langle u|v \rangle - 1_K(v) \} ,$$

et donc le sup se restreint à $v \in K$. Donc

$$J(v) = \sup_{u \in K} \langle u|v \rangle .$$

1.2 Utilisation de la transformée de Legendre-Fenchel pour la sous-différentielle.

Théorème 1

Soit $f \in \Gamma_0$ alors :

$$u \in \partial f(v) \iff v \in \partial f^*(u).$$

Preuve On la trouvera dans [5], page 48.

Soit $g \in \mathbb{R}^{N \times N}$ une image, $\lambda > 0$. On pose $\|u\|^2 = \langle u|u \rangle$. On veut minimiser

$$\min_{u \in X} \frac{\|u - g\|_2^2}{2\lambda} + J(u).$$

L'équation de Legendre s'écrit

$$0 \in u - g + \lambda \partial J(u),$$

où ∂J est la sous-différentielle définie plus haut.

Ce qui est équivalent, par le théorème 1, à :

$$u \in \partial J^* \left(\frac{g - u}{\lambda} \right),$$

ou

$$\frac{g}{\lambda} \in \frac{g - u}{\lambda} + \frac{1}{\lambda} \partial J^* \left(\frac{g - u}{\lambda} \right). \quad (1)$$

Théorème 2

Si $J \in \Gamma_0$ alors :

$$w = (I + s\partial J)^{-1}(v) \iff w \text{ minimise } \frac{\|w - v\|^2}{2} + sJ(w).$$

Théorème 3

Si $J^* = 1_K$, alors l'opérateur $(I + s\partial J^*)^{-1}$ est la projection sur l'ensemble convexe fermé K , notée π_K .

Preuve (de 2 et 3.)

$$\begin{aligned} w = (I + s\partial J^*)^{-1}(v) &\iff v = w + s\partial J^*(w) \\ &\iff 0 \in w - v + s\partial J^*(w) \\ &\iff w \text{ minimise } \frac{\|w - v\|^2}{2} + sJ^*(w) \end{aligned}$$

avec $J^* = 1_K$.

Avec de telle notations, 1 devient

$$u = g - \lambda \left(I + \frac{1}{\lambda} \partial J^* \right)^{-1} \left(\frac{g}{\lambda} \right).$$

Et la solution de notre problème initial devient $u = g - \pi_K(g)$.

1.3 Application de la transformation de Legendre-Fenchel à la minimisation de la variation totale.

Définition

Posons

$$\nabla u_{i,j} = \begin{pmatrix} \begin{cases} u_{i+1,j} - u_{i,j} & \text{si } i < N, \\ 0 & \text{si } i = N, \\ u_{i,j+1} - u_{i,j} & \text{si } j < N, \\ 0 & \text{si } j = N. \end{cases} \end{pmatrix}.$$

On pose également :

$$\operatorname{div}(p)_{i,j} = \begin{cases} p_{i,j}^1 - p_{i-1,j}^1 & \text{si } 1 < i < N, \\ p_{i,j}^1 & \text{si } i = 1, \\ -p_{i-1,j}^1 & \text{si } i = N, \end{cases} + \begin{cases} p_{i,j}^2 - p_{i,j-1}^2 & \text{si } 1 < j < N, \\ p_{i,j}^2 & \text{si } j = 1, \\ -p_{i,j-1}^2 & \text{si } j = N, \end{cases}.$$

Lemme 1 (Dualité gradient-divergence.)

$$\langle -\operatorname{div} p | u \rangle = \langle p | \nabla u \rangle.$$

La démonstration est uniquement calculatoire et tombe juste car la divergence a été définie pour cela.

Cette formule est l'équivalent discret d'une formule découlant de la formule de Stokes appelée parfois formule du gradient. Soit $f \in C^1$, \vec{A} un champ de vecteur :

$$\int_{\mathbb{R}^n} f \operatorname{div} \vec{A} d\lambda_n(x) = - \int_{\mathbb{R}^n} \nabla f \cdot \vec{A} d\lambda_n(x)$$

On veut minimiser

$$J(u) = \sum_{1 \leq i,j \leq N} \sqrt{\left(\nabla u_{i,j}^{(1)}\right)^2 + \left(\nabla u_{i,j}^{(2)}\right)^2}. \quad (2)$$

Pour minimiser $J(u)$ on utilise le résultat suivant :

Proposition 3

$$\begin{aligned} J(u) &:= \sum_{1 \leq i,j \leq N} \sqrt{\left(\nabla u_{i,j}^{(1)}\right)^2 + \left(\nabla u_{i,j}^{(2)}\right)^2} \\ &= \sup_{\{p \text{ tq } |p_{i,j}| \leq 1\}} \langle p | \nabla u \rangle. \end{aligned}$$

Preuve En effet, on a par Cauchy-Schwarz :

$$p_{i,j}^{(1)} \nabla u_{i,j}^{(1)} + p_{i,j}^{(2)} \nabla u_{i,j}^{(2)} \leq |\nabla u_{i,j}| |p_{i,j}|,$$

et si $|p_{i,j}| = 1$, on a

$$p_{i,j}^{(1)} \nabla u_{i,j}^{(1)} + p_{i,j}^{(2)} \nabla u_{i,j}^{(2)} \leq |\nabla u_{i,j}|,$$

d'où

$$\sum_{i,j} p_{i,j}^{(1)} \nabla u_{i,j}^{(1)} + p_{i,j}^{(2)} \nabla u_{i,j}^{(2)} \leq \sum_{i,j} |\nabla u_{i,j}|$$

et

$$\sup_{\{p \text{ tq } |p_{i,j}| \leq 1\}} \sum_{i,j} p_{i,j}^{(1)} \nabla u_{i,j}^{(1)} + p_{i,j}^{(2)} \nabla u_{i,j}^{(2)} \leq \sum_{i,j} |\nabla u_{i,j}|.$$

Et on obtient le cas d'égalité, si $|\nabla u_{i,j}| \neq 0$ grâce à $p_{i,j} = \frac{\nabla u_{i,j}}{|\nabla u_{i,j}|}$. Si $|\nabla u_{i,j}| = 0$, l'égalité est immédiatement vérifiée.

Corollaire 3

On déduit du lemme 1 que

$$J(u) = \sup_{\{\operatorname{div} p \text{ tq } |p_{i,j}| \leq 1, \forall i,j=1,\dots,N\}} \langle p | u \rangle.$$

1.4 Application au débruitage d'image.

1.4.1 Considérations théoriques en vue de la construction de l'algorithme.

Le modèle ROF tient son nom de celui de ses inventeurs : Rudin, Osher et Fatemi : Soit $g \in \mathbb{R}^{N \times N}$ une image, $\lambda > 0$. On pose $\|u\|^2 = \langle u|u \rangle$. On veut minimiser

$$\min_{u \in X} \frac{\|u - g\|_2^2}{2\lambda} + J(u) ,$$

avec J la variation totale dont on rappelle l'expression :

$$J(u) = \sum_{1 \leq i,j \leq N} \sqrt{\left(\nabla u_{i,j}^{(1)}\right)^2 + \left(\nabla u_{i,j}^{(2)}\right)^2}.$$

D'après le travail fait dans 1.2 , la solution de notre problème est sous la forme d'une projection $u = g - \pi_K(g)$.

Chambolle, dans son article [1], explique qu'il est facile de programmer cette projection en dimension 1. Et il propose un algorithme pour calculer la projection en dimension 2. Il s'agit de résoudre le problème suivant :

$$\min \left\{ \|\lambda \operatorname{div} p - g\|^2 \text{ sous contrainte } \|p_{i,j}\|_2^2 \leq 1 \right\}.$$

Les relations de Karush-Kuhn-Tucker s'écrivent ainsi : Il existe $\alpha_{i,j}$ tels que :

$$\begin{cases} \nabla(\lambda \operatorname{div} p - g)_{i,j} + \alpha_{i,j} p_{i,j} = 0 \\ \alpha_{i,j} \geq 0 \\ \alpha_{i,j} (\|p_{i,j}\|_2^2 - 1) = 0 \end{cases}.$$

On a deux éventualités : $\alpha_{i,j} > 0$ et $\|p_{i,j}\|_2 = 1$; ou bien $\alpha_{i,j} = 0$ et $\|p_{i,j}\|_2 < 1$. Dans les deux cas $\alpha_{i,j} = |\nabla(\lambda \operatorname{div} p - g)_{i,j}|$. On obtient ainsi l'équation à point fixe :

$$p_{i,j}^{(n+1)} = p_{i,j}^{(n)} + \tau \left((\nabla(\lambda \operatorname{div} p - g))_{i,j} - |\nabla(\lambda \operatorname{div} p - g)_{i,j}| p_{i,j}^{(n+1)} \right)$$

avec $\tau > 0$. Et de façon explicite :

$$p_{i,j}^{(n+1)} = \frac{p_{i,j}^{(n)} + \tau(\nabla(\lambda \operatorname{div} p^{(n)} - g))_{i,j}}{1 + \tau|\nabla(\lambda \operatorname{div} p^{(n)} - g)_{i,j}|}.$$

La convergence de cet algorithme est démontré dans [1].

Si l'on considère l'algorithme tel quel, on peut faire du débruitage en projetant l'image bruitée sur K .

1.5 L'algorithme de Forward-Backward.

1.5.1 Cadre et présentation : le point de vue proximal.

Le cadre d'application de l'algorithme de Forward-Backward est de manière générale la minimisation de problème d'optimisation convexe de la forme :

$$J(v_0) = \min_{v \in \mathbb{R}^n} f_1(v) + f_2(v).$$

On a un algorithme, dit de **Forward-Backward**, donné par :

$$x_{n+1} \leftarrow \pi_{\lambda K} (x - \gamma_n \nabla f_1(x_n)).$$

1.5.2 Convergence.

Théorème 4

Soient $f_0 \in \Gamma_0$ et $f_1 : \mathbb{R}^N \rightarrow \mathbb{R}$ convexe et différentiable telle que :

$$\forall x, y \in \mathbb{R}^N, \|\nabla f_1(x) - \nabla f_1(y)\| \leq \beta \|x - y\|,$$

où $\beta > 0$. On suppose aussi que $f_0 + f_1$ est coercive. Alors il est montré dans [2] que le problème

$$J(v_0) = \min_{v \in \mathbb{R}^n} f_0(v) + f_1(v)$$

admet une unique solution caractérisée par l'équation à point fixe

$$x = \text{prox}_{\lambda f_0}(x - \gamma \nabla f_1(x)).$$

1.5.3 Implémentation.

On utilisera ici la projection utilisée dans la section ??, que l'on avait appelée

```
function Im=chambolle_simple( N,lambda,g)
```

Notre algorithme de Forward-Backward sera écrit de la manière suivante :

```
for itt=1:N
    v=v-gamma*(v-g)/lambda;
    v=chambolle_simple(iterations_gradient,lambda,v);
end
```

1.6 Application à la déconvolution.

On veut déconvoluer en minimisant le critère :

$$\min_{u \in X} \frac{\|T * u - g\|_2^2}{2\lambda} + J(u). \quad (3)$$

On a donc un algorithme donné par Forward-Backward car le premier terme est dérivable :

$$x_{n+1} \leftarrow \pi_{\lambda K}(x - \gamma_n \nabla f_1(x_n)).$$

On détermine $\nabla f_1(x_n)$:

$$\begin{aligned} \nabla f_1(u) &= \nabla [\|Tu - g\|_2^2] \\ &= \nabla [u^t T^t T u - 2u^t T^t g + \|g\|_2^2] \\ &= 2T^t T u - 2T^t g \end{aligned}$$

L'algorithme est donné par :

$$x_{n+1} \leftarrow \pi_{\lambda K}(x - \gamma_n(2T^t T x_n - 2T^t g)).$$

1.7 Application à l'inpainting.

On veut utiliser l'algorithme de Forward-Backward pour réaliser l'inpainting d'une image en cherchant à minimiser la *variation totale*. On choisit donc de le construire avec le formalisme des opérateurs proximaux. Pour réaliser cela on explicite les différentes projections qui vont intervenir. Il y a ici, en l'occurrence, 2. La première projection est celle qui minimise la variation totale de l'image, donnée par Chambolle dans [1], et que l'on a noté jusqu'à présent $\pi_{\lambda K}$. La deuxième projection sera celle qui minimise la distance entre l'image masquée et l'image courante de l'algorithme. On donne ainsi :

$$P_2(v_{i,j}) = \begin{cases} v_{i,j} & \text{si le masque est à valeur 0} \\ g_{i,j} & \text{sinon} \end{cases}.$$

C'est une projection sur un sous-espace vectoriel (par conséquent, un ensemble fermé et convexe) d'un espace hilbertien. L'algorithme de Forward-Backward deviendra alors :

$$x_{n+1} \leftarrow \pi_{\lambda K} (P_2(x_n)).$$

On peut aussi augmenter l'efficacité de l'algorithme en lui insérant la méthode de Beck-Téboulle.

2 Utilisation d'algorithmes "primal-dual" pour des problèmes non lisses.

2.1 Le critère TV- L^1 . Résolution par algorithme primal-dual.

2.1.1 Mise en place théorique de l'algorithme primal-dual.

On veut minimiser le critère :

$$\min_{u \in X} \lambda \|u - g\|_1 + J(u), \text{ avec :}$$

$$J(u) = \sum_{1 \leq i, j \leq N} \sqrt{(\nabla u_{i,j}^1)^2 + (\nabla u_{i,j}^2)^2}.$$

On pose le problème dual associé² :

$$\min_{u \in X} \max_{p \in Y} [-\langle u | \text{div} p \rangle_X + \lambda \|u - g\|_1 - \delta_P(p)] , \quad (4)$$

avec

$$\delta_P(p) = \begin{cases} 0 & \text{si } p \in P, \\ +\infty & \text{sinon.} \end{cases}$$

Et $P = \{p \in Y : \|p\|_\infty \leq 1\}$.

Ce problème est d'une forme que l'on identifie aisément à la forme généralisée :

$$\min_{u \in X} \max_{p \in Y} [\langle Ku | p \rangle_X + G(u) - F^*(y)] , \quad (5)$$

associé au problème dit "primal" :

$$\min_{u \in X} [F(Kx) + G(x)] ,$$

où $G : X \leftarrow [0, +\infty]$; $F^* : Y \leftarrow [0, +\infty]$ est propre, semi-continue inférieurement, et convexe; K est un opérateur linéaire continu.

Un algorithme de résolution de base, est donné dans [6] :

Algorithme : 1

- On initialise $\tau, \sigma > 0, \theta \in [0, 1]$, $(x^0, y^0) \in X \times Y$ et $\bar{x}^0 = x^0$.
- On itère :
 - $y^{n+1} = (I + \sigma \partial F^*)^{-1}(y^n + \sigma K \bar{x}^n)$
 - $x^{n+1} = (I + \tau \partial G)^{-1}(x^n + \tau K^* y^{n+1})$
 - $\bar{x}^n = x^{n+1} + \theta(x^{n+1} - x^n)$.

Ce que l'on peut écrire de la façon suivante :

On itère :

- $y^{n+1} = \text{prox}_{\sigma F^*}(y^n + \sigma K \bar{x}^n)$
- $x^{n+1} = \text{prox}_{\tau G}(x^n + \tau K^* y^{n+1})$
- $\bar{x}^n = x^{n+1} + \theta(x^{n+1} - x^n)$.

2. Voir pour cela [?] ou [6] page 2.

2.2 Inpainting par minimisation de la variation totale : approche par algorithme primal-dual.

On définit le modèle

$$\min_{u \in X} J(u) + \frac{\lambda}{2} \sum_{i,j \in D \setminus I} (u_{i,j} - g_{i,j})^2 ,$$

où I est le domaine des données perdues (domaine d'inpainting³)

On réécrit le problème dual :

$$\min_{u \in v} \max_{p \in Y} \langle \nabla u | p \rangle + \frac{\lambda}{2} \sum_{i,j \in D \setminus I} (u_{i,j} - g_{i,j})^2 + \delta_P(p).$$

De même qu'avant, identifions dans 4 les éléments de 5 afin de pouvoir appliquer l'algorithme 1 :

$$Kx = \nabla(x) ,$$

$$F^* = \delta_P(p) ,$$

$$G = \frac{\lambda}{2} \sum_{i,j \in D \setminus I} (u_{i,j} - g_{i,j})^2 .$$

3. Les "trous à boucher".

Références

- [1] Antonin Chambolle *An algorithm for Total Variation Minimisation and Application*,
- [2] Combettes, Wajs, *Signal recovery by proximal forward-backward splitting. Multiscale Model.*, 1168-1200 (2005).
- [3] Combettes, Pesquet, *Proximal Splitting Methods in Signal Processing*.
- [4] Jean-Baptiste Hiriart-Urruty et Claude Lemaréchal, *Convex Analysis and Minimization algorithms 1.*, Springer-Verlag, 1993.
- [5] Jean-Baptiste Hiriart-Urruty et Claude Lemaréchal, *Convex Analysis and Minimization algorithms 2.*, Springer-Verlag, 1993.
- [6] Antonin Chambolle, Thomas Pock, *A first-order primal-dual algorithm for convex problems with applications to imaging*, hal-00490826-v1, 9/06/2010.
- [7] Fabien PIERRE, *Optimisation pour les méthodes variationnelles appliquées au traitement d'image, rapport de stage* , 2012.