Convex nonsmooth optimization
Part III: Algorithms

Barbara Pascal

LS2N, CNRS, Centrale Nantes, Nantes University, Nantes, France
barbara.pascal@cnrs.fr

Collaboration

This course has been built by
Jean-Christophe Pesquet (CentraleSupélec) and Nelly Pustelnik (LPENSL)

Optimization algorithm: *Forward-Backward*

Let $\mathcal{H}$ be a Hilbert space.
Let $f \in \Gamma_0(\mathcal{H})$.
Let $g \in \Gamma_0(\mathcal{H})$ be differentiable with a ν-Lipschitzian gradient where $\nu \in]0, +\infty[$.
Let $\gamma \in]0, 2/\nu[$ and $\delta = \min\{1, 1/(\nu\gamma)\} + 1/2$.
Let $(\lambda_n)_{n\in\mathbb{N}}$ be a sequence in $[0, \delta[$ such that $\sum_{n\in\mathbb{N}} \lambda_n(\delta - \lambda_n) = +\infty$.
We assume that $\mathrm{Argmin}(f + g) \neq \varnothing$. Let $x_0 \in \mathcal{H}$ and

$$(\forall n \in \mathbb{N}) \qquad \begin{cases} y_n = x_n - \gamma \nabla g(x_n) \\ x_{n+1} = x_n + \lambda_n(\mathrm{prox}_{\gamma f} y_n - x_n). \end{cases}$$

Then, $(x_n)_{n\in\mathbb{N}}$ converges weakly to a minimizer of $f + g$.

Optimization algorithm: projected gradient

Let $\mathcal{H}$ be a Hilbert space.
Let C a nonempty closed convex subset of $\mathcal{H}$.
Let $g \in \Gamma_0(\mathcal{H})$ be differentiable with a ν-Lipschitzian gradient where $\nu \in]0, +\infty[$.
Let $\gamma \in]0, 2/\nu[$ and $\delta = \min\{1, 1/(\nu\gamma)\} + 1/2$.
Let $(\lambda_n)_{n\in\mathbb{N}}$ be a sequence in $[0, \delta[$ such that $\sum_{n\in\mathbb{N}} \lambda_n(\delta - \lambda_n) = +\infty$.
We assume that $\mathrm{Argmin}_{x\in C}\, g(x) \neq \varnothing$. Let $x_0 \in \mathcal{H}$ and

$$(\forall n \in \mathbb{N}) \qquad \begin{cases} y_n = x_n - \gamma \nabla g(x_n) \\ x_{n+1} = x_n + \lambda_n(P_C y_n - x_n). \end{cases}$$

Then, $(x_n)_{n\in\mathbb{N}}$ converges weakly to a minimizer of g over C.

Optimization algorithm: gradient descent

Let $\mathcal{H}$ be a Hilbert space.
Let $g \in \Gamma_0(\mathcal{H})$ be a differentiable function with a ν-lipschitzian gradient where $\nu \in\ ]0, +\infty[$.
Le $\gamma \in]0, 2/\nu[$.
We assume that $\operatorname{Argmin} g \neq \varnothing$. Let $x_0 \in \mathcal{H}$ and

$$(\forall n \in \mathbb{N}) \qquad x_{n+1} = x_n - \gamma \nabla g(x_n)$$

Then, $(x_n)_{n\in\mathbb{N}}$ converges weakly to a minimizer of f.

Optimization algorithm: Douglas-Rachford

Let $\mathcal{H}$ be a Hilbert space.

Let $f \in \Gamma_0(\mathcal{H})$ and $g \in \Gamma_0(\mathcal{H})$.

$$(\forall n \in \mathbb{N}) \qquad \begin{cases} y_n = \operatorname{prox}_{\gamma g} x_n \\ z_n = \operatorname{prox}_{\gamma f}(2y_n - x_n) \\ x_{n+1} = x_n + \lambda_n(z_n - y_n). \end{cases}$$

Optimization algorithm: Douglas-Rachford

Let $\mathcal{H}$ be a Hilbert space.

Let $f \in \Gamma_0(\mathcal{H})$ and $g \in \Gamma_0(\mathcal{H})$.

Let $\gamma \in \,]0,+\infty[$ and let $(\lambda_n)_{n\in\mathbb{N}}$ be a sequence in $[0,2]$ such that $\sum_{n\in\mathbb{N}} \lambda_n(2-\lambda_n) = +\infty$.

We assume that $\operatorname{zer}(\partial f + \partial g) \neq \varnothing$. Let $x_0 \in \mathcal{H}$ and

$$(\forall n \in \mathbb{N}) \qquad \begin{cases} y_n = \operatorname{prox}_{\gamma g} x_n \\ z_n = \operatorname{prox}_{\gamma f}(2y_n - x_n) \\ x_{n+1} = x_n + \lambda_n(z_n - y_n). \end{cases}$$

The following properties are satisfied:

- $x_n \rightharpoonup \widehat{x}$
- $z_n - y_n \to 0$, $y_n \rightharpoonup \widehat{y}$, $z_n \rightharpoonup \widehat{y}$ where $\widehat{y} = \operatorname{prox}_{\gamma g}\widehat{x} \in \operatorname{Argmin}(f+g)$.

Optimization algorithm: Douglas-Rachford

Let $\mathcal{H}$ and $\mathcal{G}$ be two Hilbert spaces.
Let $g \in \Gamma_0(\mathcal{H})$ and $L \in \mathcal{B}(\mathcal{G}, \mathcal{H})$ such that $\operatorname{ran} L$ (image) is closed and $L^* L$ is a isomorphism.
Let $\gamma \in \,]0, +\infty[$ and let $(\lambda_n)_{n\in\mathbb{N}}$ a sequence in $[0, 2]$ such that $\sum_{n\in\mathbb{N}} \lambda_n(2-\lambda_n) = +\infty$.
We assume that $\operatorname{zer}(L^* \circ \partial g \circ L) \neq \varnothing$. Let $x_0 \in \mathcal{H}$, $v_0 = (L^*L)^{-1} L^* x_0$ et

$$(\forall n \in \mathbb{N}) \qquad \begin{cases} y_n = \operatorname{prox}_{\gamma g} x_n \\ c_n = (L^*L)^{-1} L^* y_n \\ x_{n+1} = x_n + \lambda_n \big(L(2c_n - v_n) - y_n \big) \\ v_{n+1} = v_n + \lambda_n (c_n - v_n). \end{cases}$$

We have then:
$v_n \rightharpoonup \widehat{v}$ where $\widehat{v} \in \operatorname{Argmin}(g \circ L)$.

Optimization algorithm: Douglas-Rachford

Sketch of proof:

$$\underset{v \in \mathcal{G}}{\text{minimize}} \;\; g(Lv) \quad \Leftrightarrow \quad \underset{x \in \mathcal{H}}{\text{minimize}} \;\; \iota_E(x) + g(x)$$

where $E = \operatorname{ran} L$.
We apply Douglas-Rachford algorithm with
$f = \iota_E \Rightarrow \operatorname{prox}_{\gamma f} = P_E$ by setting

$$(\forall n \in \mathbb{N}) \qquad P_E y_n = Lc_n \;\text{ and }\; P_E x_n = Lv_n$$

where $c_n = \underset{c \in \mathcal{H}}{\operatorname{argmin}} \;\; \|y_n - Lc\|^2 = (L^* L)^{-1} L^* y_n$.

Optimization algorithm: Douglas-Rachford

Particular case of Douglas-Rachford algorithm:
$\mathcal{H} = \mathcal{H}_1 \times \cdots \mathcal{H}_m$ where $\mathcal{H}_1, \ldots, \mathcal{H}_m$ Hilbert spaces
$\big(\forall x = (x_1, \ldots, x_m) \in \mathcal{H}\big)\ g(x) = \sum_{i=1}^m g_i(x_i)$
where $(\forall i \in \{1, \ldots, m\}\ g_i \in \Gamma_0(\mathcal{H}_i)$
$L\colon v \mapsto (L_1 v, \ldots, L_m v)$ where $(\forall i \in \{1, \ldots, m\})\ L_i \in \mathcal{B}(\mathcal{G}, \mathcal{H}_i)$.

PPXA+ algorithm

Let $(x_{0,i})_{1 \le i \le m} \in \mathcal{H},\ v_0 = (\sum_{i=1}^m L_i^* L_i)^{-1} \sum_{i=1}^m L_i^* x_{0,i}$ and

$$(\forall n \in \mathbb{N}) \quad \begin{cases} y_{n,i} = \operatorname{prox}_{\gamma g_i} x_{n,i}, \quad i \in \{1, \ldots, m\} \\ c_n = (\sum_{i=1}^m L_i^* L_i)^{-1} \sum_{i=1}^m L_i^* y_{n,i} \\ x_{n+1,i} = x_{n,i} + \lambda_n \big(L_i(2c_n - v_n) - y_{n,i}\big), \quad i \in \{1, \ldots, m\} \\ v_{n+1} = v_n + \lambda_n (c_n - v_n). \end{cases}$$

We have then $v_n \rightharpoonup \widehat{v} \in \operatorname{Argmin} \sum_{i=1}^m g_i \circ L_i$.

Optimization algorithm: Douglas-Rachford

Particular case of Douglas-Rachford algorithm:

$\mathcal{H} = \mathcal{H}_1 \times \cdots \mathcal{H}_m$ where $\mathcal{H}_1 = \ldots = \mathcal{H}_m$ Hilbert spaces

$\big(\forall x = (x_1, \ldots, x_m) \in \mathcal{H}\big)\ g(x) = \sum_{i=1}^m g_i(x_i)$

where $(\forall i \in \{1, \ldots, m\}\ g_i \in \Gamma_0(\mathcal{H}_i)$

$L\colon v \mapsto (L_1 v, \ldots, L_m v)$ where $L_1 = \ldots = L_m = \mathrm{Id}.$

PPXA algorithm

Let $(x_{0,i})_{1 \leq i \leq m} \in \mathcal{H}$, $v_0 = \frac{1}{m} \sum_{i=1}^m x_{0,i}$ and

$$(\forall n \in \mathbb{N}) \qquad \begin{cases} y_{n,i} = \mathrm{prox}_{\gamma g_i} x_{n,i}, \quad i \in \{1, \ldots, m\} \\ c_n = \frac{1}{m} \sum_{i=1}^m y_{n,i} \\ x_{n+1,i} = x_{n,i} + \lambda_n \big(2c_n - v_n - y_{n,i}\big), \quad i \in \{1, \ldots, m\} \\ v_{n+1} = v_n + \lambda_n (c_n - v_n). \end{cases}$$

We have then $v_n \rightharpoonup \widehat{v} \in \mathrm{Argmin} \sum_{i=1}^m g_i$.

Optimization algorithms

Forward-Backward	$f_1 + f_2$	f_1 gradient Lipschitz prox_{f_2}	[Combettes,Wajs,2005]
ISTA	$f_1 + f_2$	f_1 gradient Lipschitz $f_2 = \lambda \Vert \cdot \Vert_1$	[Daubechies et al, 2003]
Projected gradient	$f_1 + f_2$	f_1 gradient Lipschitz $f_2 = \iota_C$	
Gradient descent	$f_1 + f_2$	f_1 gradient Lipschitz $f_2 = 0$	
Douglas-Rachford	$f_1 + f_2$	prox_{f_1} prox_{f_2}	[Combettes,Pesquet, 2007]
PPXA	$\sum_i f_i$	prox_{f_i}	[Combettes,Pesquet, 2008]
PPXA+	$\sum_i f_i \circ L_i$	prox_{f_i} $(\sum_{i=1}^m L_i^* L_i)^{-1}$	[Pesquet, Pustelnik, 2012]