Proximal Point Algorithm and Method of Multipliers

6:37 PM

Proximal_point_and_...

def: monotone operators definitions and related

page1 Damped Cayley Iteration: Consider F{monotone}

$F(x)\ni 0$ solve করতে হবে

$\updownarrow$

$C(x)=x$ Why find fixed point of Cayley and resolvent of some monotone operator?

[# damped iteration for finding fixed point of nonexpansive mapping #]

now as C is a nonexpansive operator, so $x^{k+1}=C(x^k)$ might not converge, but damped iteration will work

{ (\lambda>=0, F monotone) => C [* nonexpansive function*] }

$$x^{k+1}=\theta C(x^k)+(1-\theta)x^k \quad \theta\in(0,1)$$

[# $C(\boxminus)=2R(\boxminus)-I\boxminus$, $C(x^k)=2R(x^k)-x^k$ #]

$$=\theta(2R(x^k)-x^k)+x^k-\theta x^k$$

$$=2\theta R(x^k)-2\theta x^k+x^k$$

$$=\underbrace{2\theta}_{\eta} R(x^k)+\underbrace{(1-2\theta)}_{\eta}x^k$$

$$=\eta R(x^k)+(1-\eta)x^k \quad [\# \eta=2\theta\in(0,2)]$$

this will converge! see the proof at [# damped iteration for finding fixed point of nonexpansive

When, $\eta=1$, then:

$$x^{k+1}=\eta R(x^k)+(1-\eta)x^k=R(x^k)+0\,x^k=R(x^k)$$ // This is also called the // proximal point algorithm, when $F=\partial f$ then thm: proximal operator is the resolvent of subdifferential operator

$\therefore \boxed{x^{k+1}=R(x^k)}$ {proximal point method}

This is saying something very important, if C is nonexpansive then x^{k+1}=R(x^k) will converge, as this is the damped Cayley iteration for specific parameter value!!! Orthat, amra bolte pari, if F is monotone => C is nonexpansive then 0 \in F(x) can be solved by solving R(x)=x, inspite of R being nonexpansive. This is a very special case, as for general nonexpansive operator O, x^{k+1}=O(x^{k}) may not converge, but when O=R it does!!! (I need to ensure this observation in future!)

turns into the proximal gradient algorithm

$F=\partial f$, $\partial\{\square\}$ will converge!

0 \in F(x) <=> R(x)=x

$$x^{k+1}=R(x)=\mathrm{prox}_{f,\lambda}(x^k)=\underset{x}{\mathrm{argmin}}\left(f(x)+\frac{1}{2\lambda}\|x-x^k\|_2^2\right)$$

*Multiplier to residual mapping: (MRM mapping) def: multiplier to residual mapping , # resolvent for multiplier to residual mapping

By definition,

$$F(\boxminus)=b-A\,\underset{x}{\mathrm{argmin}}\,L(x,\boxminus)$$ {this is a monotone operator}

$$L(x,v)=f(x)+v^T(Ax-b)$$

$$\begin{pmatrix} \vee & f(x) \\ \urcorner & Ax=b \end{pmatrix}$$ # The optimization problem in consideration

$$\therefore\ 0\in F(v)=b-A\,\underbrace{\underset{x}{\mathrm{argmin}}\,L(x,v)}$$

will produce $x^*(v)$ optimal primal variable as a function of v

so a $0\in F(v^*)$ means we also have the optimal dual variable,

so, $x^*(v^*)=x^*$ will be the optimal primal variable.

Why find fixed point of Cayley and resolvent of some monotone operator?

now from the resolvent-monotone operator relation থেকে জানি:

$R(v)=v$ will give the v^*, as MRM is monotone, and for $\lambda\geq 0$. its resolvent is nonxpansive

পাবো,

resolvent for multiplier to residual mapping থেকে জানি

$\therefore R(y)=z$ can be determined from:

$\therefore R(y) = z$ can be determined from:

$$x := \underset{w}{\operatorname{argmin}} \left(f(w) + y^T(Aw-b) + \left(\frac{\lambda}{2}\right) \|Aw-b\|_2^2 \right)$$

$$z := y + \lambda(Ax - b)$$

$*$ now when we are interested in finding the fixed point of R. say $R(y^*) = y^*$ then we can use the iteration:

$R(y^k) = y^{k+1}$

Because, if F:monotone, $\lambda \geq 0$ then $R=(I+\lambda F)^{-1}$:nonexpansive
As resolvent is an exceptional nonexpansive function that does not require damped iteration, normal contraction like iteration is sufficient to find the fixed point. So, we can use $x^{k+1}=R(x^k)$ to find the fixed point.

$y := y^k$
$z := y^{k+1}$
$x := x^{k+1}$ # this is for just keeping track

বাকিটা হাই:

$$x^{k+1} := \underset{w}{\operatorname{argmin}} \left(f(w) + y^T(Aw-b) + \frac{\lambda}{2} \|Aw-b\|_2^2 \right)$$

$$y^{k+1} := y^k + \lambda(Ax^{k+1} - b)$$

(old explanation, can be ignored)

MRM এর resolvent এর fixedpoint বের করা যায় by using the following formula:

$R(y) = y$

$$x = \underset{w}{\operatorname{argmin}} \left(f(w) + y^T(Aw-b) + \frac{\lambda}{2} \|Aw-b\|_2^2 \right)$$

$$y = y + \lambda(Ax - b)$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \underbrace{\begin{bmatrix} \underset{w}{\operatorname{argmin}} \; f(w) + \square^T(Aw-b) + \frac{\lambda}{2}\|Aw-b\|_2^2 \\ \square + \lambda(A\square - b) \end{bmatrix}}_{\left(R\left(\begin{smallmatrix} \square \\ \square \end{smallmatrix}\right)\right)} \begin{bmatrix} x \\ y \end{bmatrix}$$

here $\begin{bmatrix} \square \\ \square \end{bmatrix}$ is the structure of the input argument of the resolvent

// যদিও technically আমরা y বের করতে চাই. y বের করতে গেলে
// $x = \operatorname{argmin}(\square)$ করতে হবে, so R is essentially is a block
// operator operating on both x and y, and we have already
// proven MRM is monotone, hence for $\lambda \geq 0$, its resolvent is
// nonexpansive.

So, by using proximal point algorithm: $\begin{bmatrix} x^{k+1} \\ y^{k+1} \end{bmatrix} = \left(R\begin{pmatrix} \square \\ \square \end{pmatrix}\right) \begin{bmatrix} x^k = \square \\ y^k = \square \end{bmatrix}$

$$\begin{bmatrix} x^{k+1} \\ y^{k+1} \end{bmatrix} = \begin{pmatrix} \underset{w}{\operatorname{argmin}} \; f(w) + \square^T(Aw-b) + \frac{\lambda}{2}\|Aw-b\|_2^2 \\ \square + \lambda(A\square - b) \end{pmatrix} \begin{bmatrix} x^k \\ y^k \end{bmatrix}$$

$$\Leftrightarrow \begin{bmatrix} x^{k+1} \\ y^{k+1} \end{bmatrix} = \begin{bmatrix} \underset{w}{\operatorname{argmin}} \; f(w) + (y^k)^T(Aw-b) + \frac{\lambda}{2}\|Aw-b\|_2^2 \\ y^k + \lambda(Ax^k - b) \end{bmatrix}$$

Method of multipliers

$$x^{k+1} = \underset{w}{\operatorname{argmin}} \; f(w) + (y^k)^T(Aw-b) + \frac{\lambda}{2}\|Aw-b\|_2^2$$

$$y^{k+1} = y^k + \lambda(Ax^k - b)$$

Will converge to primal dual pair

// if $f(\square)$ is not strictly convex, then '=' এর পরিবর্তে '$\in$' ব্যবহার করবো.

eq: method of multipliers

So, primal feasibility occurs as $k \to \infty$

the first stage is augmented Lagrangian minimization, $x^k \hookrightarrow x^*$, so $Ax^k - b \hookrightarrow 0$

the second stage is update the dual variable, $y^k \hookrightarrow y^*$

- the second stage is update the dual variable, $y^k \hookrightarrow y^{[illegible]}$

Method of multipliers, though like dual method,

- uses augmented Lagrangian
- converges under far more general conditions (than dual method)
- f can be non differentiable, can be $+\infty$
- disadvantage: in this form not easy to decompose.