(Fast) Primal-dual proximal gradient algorithm and preconditioning

Andersen Ang

Mathématique et recherche opérationnelle
UMONS, Belgium

manshun.ang@umons.ac.be Homepage: angms.science

First draft : March 31, 2020
Last update : May 10, 2020

Problem setting

$$(P) \quad \min_{\mathbf{x}} f(\mathbf{x}) + g(\mathbf{A}\mathbf{x})$$

- $f : \mathbb{R}^n \to \mathbb{R}$, $g : \mathbb{R}^n \to \bar{\mathbb{R}}$.
- $\bar{\mathbb{R}} = \mathbb{R} \cup \{\pm\infty\}$ is the extended real line.
- f is proper, close and strongly convex.
- g is proper, close and convex, possibly nonsmooth.
- $\mathbf{A} \in \mathbb{R}^{m \times n}$ has full rank.
- By properties of f and g, problem (P) has a unique optimal sol. denoted by $\mathbf{x}^*$.

Deriving the dual formulation of (P) ... 1/2

- (P) written in constrained form

$$(P') \quad \min_{\mathbf{x}} f(\mathbf{x}) + g(\mathbf{z}) \text{ s.t. } \mathbf{A}\mathbf{x} = \mathbf{z}.$$

- Let $\boldsymbol{\nu}$ is the Lagrangian multiplier associated to the constraint in (P'), the Lagrangian is

$$L(\mathbf{x}, \mathbf{z}, \boldsymbol{\nu}) = f(\mathbf{x}) + g(\mathbf{z}) - \langle \boldsymbol{\nu}, \mathbf{z} - \mathbf{A}\mathbf{x} \rangle,$$

- Recall minimizing the Lagrangian w.r.t. $\mathbf{x}$ and $\mathbf{z}$ gives the dual problem. Now we split the Lagrangian into two parts : on $\mathbf{x}$ and on $\mathbf{z}$

$$\begin{aligned} L(\mathbf{x}, \mathbf{z}, \boldsymbol{\nu}) &= f(\mathbf{x}) + \langle \boldsymbol{\nu}, \mathbf{A}\mathbf{x} \rangle + g(\mathbf{z}) - \langle \boldsymbol{\nu}, \mathbf{z} \rangle. \\ \min_{\mathbf{x},\mathbf{z}} L &= \min_{\mathbf{x}} \Big\{ f(\mathbf{x}) + \langle \boldsymbol{\nu}, \mathbf{A}\mathbf{x} \rangle \Big\} + \min_{\mathbf{z}} \Big\{ g(\mathbf{z}) - \langle \boldsymbol{\nu}, \mathbf{z} \rangle \Big\} \\ &= \min_{\mathbf{x}} \Big\{ f(\mathbf{x}) + \langle \mathbf{A}^\top \boldsymbol{\nu}, \mathbf{x} \rangle \Big\} + \min_{\mathbf{z}} \Big\{ g(\mathbf{z}) - \langle \boldsymbol{\nu}, \mathbf{z} \rangle \Big\} \end{aligned}$$

- Now we use convex conjugate to simplify the notation : recall convex conjugate $h^*(\mathbf{u})$ of a function $h(\mathbf{u})$ is

$$h^*(\mathbf{u}) = \max_{\mathbf{v} \in \operatorname{dom} f} \big\{ \langle \mathbf{u}, \mathbf{v} \rangle - f(\mathbf{v}) \big\}.$$

Deriving the dual formulation of (P) ... 2/2

- Recall $\min f(\mathbf{x}) = -\max\{-f(\mathbf{x})\}$

$$\begin{aligned}\min_{\mathbf{x},\mathbf{z}} L &= \min_{\mathbf{x}} \Big\{ f(\mathbf{x}) + \langle \mathbf{A}^\top \boldsymbol{\nu}, \mathbf{x} \rangle \Big\} + \min_{\mathbf{z}} \Big\{ g(\mathbf{z}) - \langle \boldsymbol{\nu}, \mathbf{z} \rangle \Big\} \\ &= -\max_{\mathbf{x}} \Big\{ -f(\mathbf{x}) - \langle \mathbf{A}^\top \boldsymbol{\nu}, \mathbf{x} \rangle \Big\} - \max_{\mathbf{z}} \Big\{ \langle \boldsymbol{\nu}, \mathbf{z} \rangle - g(\mathbf{z}) \Big\} \\ &= -\max_{\mathbf{x}} \Big\{ -f(\mathbf{x}) + \langle -\mathbf{A}^\top \boldsymbol{\nu}, \mathbf{x} \rangle \Big\} - \max_{\mathbf{z}} \Big\{ \langle \boldsymbol{\nu}, \mathbf{z} \rangle - g(\mathbf{z}) \Big\}\end{aligned}$$

- Let f^* and g^* be the conjugate of f and g respectively, we have

$$\min_{\mathbf{x},\mathbf{z}} L = -f^*(-\mathbf{A}^\top \boldsymbol{\nu}) - g^*(\boldsymbol{\nu})$$

- The dual problem is thus

$$(D) \quad \max_{\boldsymbol{\nu}} q(\boldsymbol{\nu}) = -f^*(-\mathbf{A}^\top \boldsymbol{\nu}) - g^*(\boldsymbol{\nu}).$$

  In minimization form (consider the negated function),

$$(D') \quad \min_{\boldsymbol{\nu}} f^*(-\mathbf{A}^\top \boldsymbol{\nu}) + g^*(\boldsymbol{\nu}).$$

- Note : if we start with $\langle \boldsymbol{\nu}, \mathbf{A}\mathbf{x} - \mathbf{z} \rangle$ in the Lagrangian as in p.6 here, we get $\min_{\boldsymbol{\nu}} f^*(\mathbf{A}^\top \boldsymbol{\nu}) + g^*(-\boldsymbol{\nu})$. The two formulations are the same essentially.

Duality and f^* has Lipschitz gradient

- Recall Slater condition : if there exists $\mathbf{x} \in \text{ri}(\text{dom } f)$ and $\mathbf{z} \in \text{ri}(\text{dom } g)$ s.t. $\mathbf{A}\mathbf{x} = \mathbf{z}$, then strong duality holds, and optimal sol. of dual problem is attained.

- Recall f is strongly convex with modulus σ. This implies f^* has Lipschitz gradient : let $F(\boldsymbol{\nu}) = f^*(-\mathbf{A}^\top \boldsymbol{\nu})$,

$$\begin{aligned}
\|\nabla F(\mathbf{x}) - \nabla F(\boldsymbol{\nu})\| &= \|\mathbf{A}\nabla f^*(-\mathbf{A}^\top \mathbf{x}) - \mathbf{A}\nabla f^*(-\mathbf{A}^\top \boldsymbol{\nu})\| \\
&\leq \|\mathbf{A}\| \cdot \|\nabla f^*(-\mathbf{A}^\top \mathbf{x}) - \nabla f^*(-\mathbf{A}^\top \boldsymbol{\nu})\| \\
&\leq \|\mathbf{A}\| \cdot \tfrac{1}{\sigma} \cdot \|\mathbf{A}^\top \mathbf{x} - \mathbf{A}^\top \boldsymbol{\nu}\| \\
&\leq \|\mathbf{A}\| \cdot \tfrac{1}{\sigma} \cdot \|\mathbf{A}^\top\| \cdot \|\mathbf{x} - \boldsymbol{\nu}\| \\
&= \frac{\|\mathbf{A}\|^2}{\sigma} \|\mathbf{x} - \boldsymbol{\nu}\|
\end{aligned}$$

So $f^*(-\mathbf{A}^\top \boldsymbol{\nu})$ is $\dfrac{\|\mathbf{A}\|^2}{\sigma}$-smooth.

Solving the dual problem by proximal gradient method

- We just showed $f^*(-\mathbf{A}^\top \boldsymbol{\nu})$ is $\frac{\|\mathbf{A}\|^2}{\sigma}$-smooth.
- Hence for the dual problem

$$(D') \quad \min_{\boldsymbol{\nu}} q(\boldsymbol{\nu}) = f^*(-\mathbf{A}^\top \boldsymbol{\nu}) + g^*(\boldsymbol{\nu}),$$

it is a minimization of a sum of smooth function and a possibly nonsmooth g^*, proximal gradient method can be used[1] to solve (D').

- Accelerated proximal gradient method (e.g. FISTA) can be used to solve (D'), with convergence rate on the sequence $\{\boldsymbol{\nu}_k\}$ towards the optimal point $\boldsymbol{\nu}^*$ (exists due to duality, see last slide) as $\mathcal{O}(\frac{1}{k^2})$:

$$q(\boldsymbol{\nu}^*) - q(\boldsymbol{\nu}_k) \leq \frac{2L\|\boldsymbol{\nu}_0 - \boldsymbol{\nu}^*\|^2}{k^2} \tag{1}$$

Small note : In FISTA, it is $F(\mathbf{x}_k) - F^* \leq \frac{2L\|\mathbf{x}_0 - \mathbf{x}^*\|^2}{k^2}$, here it is $q(\boldsymbol{\nu}^*) - q(\boldsymbol{\nu}_k)$, the order is swapped since the problem on $\boldsymbol{\nu}$ is the dual problem, with the original form as (D) which is maximization not minimization. (Well, actually the sign is not important here.)

[1] If g is proximable.

The fast dual proximal gradient method

- The iterations of the fast dual proximal gradient are
  - $\boldsymbol{\nu}_k = \boldsymbol{\mu}_k + \beta_k(\boldsymbol{\mu}_k - \boldsymbol{\mu}_{k-1})$
  - $\boldsymbol{\mu}_{k+1} = \operatorname{prox}_{\frac{1}{L}g^*}\left(\boldsymbol{\nu}_k - \frac{1}{L}\nabla f^*(-\mathbf{A}^\top \boldsymbol{\nu}_k)\right)$

  where $L \geq \frac{\|\mathbf{A}\|^2}{\sigma}$, $\boldsymbol{\mu}_0 = \boldsymbol{\mu}_{-1}$, $\beta_k = \frac{k-1}{k+2}$ (slightly sub-optimal).
- Here $\boldsymbol{\nu}_k$ is the sequence of dual variable and $\boldsymbol{\mu}_k$ is the auxiliary variable for the acceleration.
- For proximal gradient step, by definition of proximal operator, we have

$$\boldsymbol{\mu}_{k+1} = \operatorname*{argmin}_{\boldsymbol{\mu}} \left\{ \frac{1}{2}\left\|\boldsymbol{\mu} - \left(\boldsymbol{\nu}_k - \frac{1}{L}\nabla f^*(-\mathbf{A}^\top \boldsymbol{\nu}_k)\right)\right\|_2^2 + \frac{1}{L}g^*(\boldsymbol{\mu}) \right\},$$

  if g is "simple" (proximable), this step can be computed readily.

Adding the primal variable

- By the definition of Lagrangian in p.3., we can define $\mathbf{x}$ from a $\boldsymbol{\nu}$ as

$$\mathbf{x}(\boldsymbol{\nu}) = \underset{\mathbf{x}}{\operatorname{argmin}}\ f(\mathbf{x}) + \langle \boldsymbol{\nu}, \mathbf{A}\mathbf{x} \rangle.$$

  Insert this into the fast dual proximal gradient method, we have
  - $\boldsymbol{\nu}_k = \boldsymbol{\mu}_k + \beta_k(\boldsymbol{\mu}_k - \boldsymbol{\mu}_{k-1})$
  - $\mathbf{x}_k = \underset{\mathbf{x}}{\operatorname{argmin}}\ f(\mathbf{x}) + \langle \boldsymbol{\nu}_k, \mathbf{A}\mathbf{x} \rangle$
  - $\boldsymbol{\mu}_{k+1} = \operatorname{prox}_{\frac{1}{L}g^*}\left(\boldsymbol{\nu}_k - \frac{1}{L}\nabla f^*(-\mathbf{A}^\top \boldsymbol{\nu}_k)\right)$
- **A trick on linking $\mathbf{x}$ to $\boldsymbol{\mu}, \boldsymbol{\nu}$.** The $\mathbf{x}$-step has no impact on $\boldsymbol{\mu}_k, \boldsymbol{\nu}_k$, this step seems to be redundant. Here is a trick : by Theorem 1 of [Richter2013][2], we have

$$\nabla f^*(-\mathbf{A}^\top \boldsymbol{\nu}) = -\mathbf{A}\mathbf{x}(\boldsymbol{\nu}),$$

  in which we can replace $\nabla f^*(-\mathbf{A}^\top \boldsymbol{\nu})$ in the $\boldsymbol{\mu}$-step as $-\mathbf{A}\mathbf{x}(\boldsymbol{\nu})$.

[2] Richter, Stefan, Colin Neil Jones, and Manfred Morari. "Certification aspects of the fast gradient method for solving the dual of parametric convex programs." Mathematical Methods of Operations Research 2013.

A (fast) primal-dual proximal algorithm

- The final algorithm reads
  - $\boldsymbol{\nu}_k = \boldsymbol{\mu}_k + \beta_k(\boldsymbol{\mu}_k - \boldsymbol{\mu}_{k-1})$
  - $\mathbf{x}_k = \underset{\mathbf{x}}{\operatorname{argmin}}\ f(\mathbf{x}) + \langle \boldsymbol{\nu}_k, \mathbf{A}\mathbf{x} \rangle$
  - $\boldsymbol{\mu}_{k+1} = \operatorname{prox}_{\frac{1}{L}g^*}\left(\boldsymbol{\nu}_k + \frac{1}{L}\mathbf{A}\mathbf{x}_k\right)$

- As we now have primal sequence in the algorithm, thus the algorithm is a kind of primal-dual algorithm.

- In general, with a general function f, the primal step ($\mathbf{x}$-step) can be difficult to solve. Hence this scheme works for the cases that the $\mathbf{x}$-step has simple close form solution, or cheap update is avaliable.

- Similarly, the dual step ($\boldsymbol{\nu}$-step) can be difficult to solve for a general g. Hence this scheme works for the cases that g is proximable.

- Notice that this scheme is in fact an accelerated primal-dual algorithm, as we have the $\boldsymbol{\nu}$-step.

Precondioning on primal variable

- We can perform precondition on primal variable to speed up the convergence
- Such precondition does not change the algorithm, this can be seen as follows. Consider the problem (P')

$$(P'_{\mathbf{E}}) \quad \min_{\mathbf{x}} f(\mathbf{x}) + g(\mathbf{z}) \text{ s.t. } \mathbf{A}\mathbf{x} = \mathbf{z},$$

We can always left multiply the equality constraint by a matrix $\mathbf{E}$ and get

$$(P'_{\mathbf{E}}) \quad \min_{\mathbf{x}} f(\mathbf{x}) + g(\mathbf{z}) \text{ s.t. } \mathbf{E}\mathbf{A}\mathbf{x} = \mathbf{E}\mathbf{z},$$

where preconditioner $\mathbf{E} \in \mathbb{R}^{m \times m}$ is a non-singular matrix.

Preconditioner

- Not any $\mathbf{E}$ can be a preconditioner. To be a preconditioner, $\mathbf{E}$ has to be non-singular, and "serves a special purpose" as follows.
- Consider the problem and the associated dual problem

$$\begin{aligned}(P'_{\mathbf{E}}) \quad & \min_{\mathbf{x}} f(\mathbf{x}) + g(\mathbf{z}) \text{ s.t. } \mathbf{E}\mathbf{A}\mathbf{x} = \mathbf{E}\mathbf{z}, \\ (D'_{\mathbf{E}}) \quad & \min_{\boldsymbol{\nu}} f^*(-\mathbf{A}^\top \mathbf{E}^\top \boldsymbol{\nu}) + g^*(\mathbf{E}^\top \boldsymbol{\nu}),\end{aligned}$$

  We can solve $(D'_{\mathbf{E}})$ by fast primal-dual proximal gradient method
  - $\boldsymbol{\nu}_k = \boldsymbol{\mu}_k + \beta_k(\boldsymbol{\mu}_k - \boldsymbol{\mu}_{k-1})$
  - $\mathbf{x}_k = \operatorname{argmin}_{\mathbf{x}} f(\mathbf{x}) + \langle \boldsymbol{\nu}_k, \mathbf{E}\mathbf{A}\mathbf{x} \rangle$
  - $\boldsymbol{\mu}_{k+1} = \operatorname{prox}_{\frac{1}{\hat{L}} g^*}\left(\boldsymbol{\nu}_k + \frac{1}{\hat{L}}\mathbf{E}\mathbf{A}\mathbf{x}_k\right)$

  where $\hat{L}$ is the Lipschitz constant of $\nabla f^*(-\mathbf{A}^\top \mathbf{E}^\top \boldsymbol{\nu})$.
- We now see that $\mathbf{E}$ affects the Lipschitz constant of ∇f^*.

Precondioner

- Recall that the Lipschitz constant of the gradient tells how circular the geometry of the level sets of the optimization problem is.
- The more circular the level sets, the less zig-zag behavior in gradient step and hence faster convergence.
- The more circulr the level sets, the closer of the Lipschitz constant L of the gradient to 1 (from above).
- Now we see that if we can find a suitable $\mathbf{E}$ such that $\hat{L}$ is as close as possible to 1, then we the preconditioner can speed up the algorithm.
- For a general function f, the constant L is related to the eigenvalue of the local Hessian $\mathbf{H} = \nabla^2 f$ of the current primal iterate, in this sense, the matrix $\mathbf{E}$ can be seen as an "eigenvalue regulator" for $\mathbf{H}$ in the iteration. And therefore, a way to find $\mathbf{E}$ is to solve a particular eigenvalue problem, which can be computationally expensive.

Last page - summary

Discussed

- Dual problem and fast dual proximal gradient method
- Adding the primal variable into dual proximal gradient method
- A (fast) Primal-dual proximal gradient algorithm
- Idea of preconditioning in the primal-dual proximal gradient method

Not discussed

- The detail convergence analysis of the primal-dual proximal gradient algorithm
- The details of preconditioning : how to actually compute $\mathbf{E}$

End of document