Bregman Method from PEP Perspective

Zhenghao Xu

Zhejiang University

October 10, 2021

Table of Contents

Progress

Finished:

- evidence of improvement (numerical)

In progress:

- optimal fixed step size (dual matrix)
- valid one-step analysis
- toolbox for explicit dual matrix (in Python)

Table of Contents

Formulation

Problem:

$$\min_{x \in \mathbb{R}^d} \quad F(x) \triangleq f(x) + \phi(x) \tag{P}$$

Assumptions:

- f is *convex*, *proper*, *lsc* and continuously differentiable.
- ϕ is *convex*, *proper*, *lsc* (possibly nonsmooth).
- h is a Legendre kernel function.
- f is L-smooth relative to h, i.e., $Lh - f$ convex.[Bauschke et al., 2017]
- f, h more restricted.
- bounded initial distance and well-posedness of problem and method.

Formulation

Problem:

$$\min_{x \in \mathbb{R}^d} \quad F(x) \triangleq f(x) + \phi(x) \tag{P}$$

Method: Bregman Proximal Gradient

$$x_{k+1} = \operatorname*{arg\,min}_{x \in \mathbb{R}^d} \{ f(x_k) + \langle \nabla f(x_k), x - x_k \rangle + \phi(x) + L D_h(x, x_k) \} \tag{BPG}$$

Case 1: h - SC & Smooth

Fix $\phi(x) \equiv 0$. Assume: h is σ_h-strongly convex and L_h-smooth. Through PEP we obtain following result:

$$(f_1 - f_*) + \frac{L\left\|\nabla h(x_1) - \nabla h(x_*) - \sqrt{L_h \sigma_h}(x_1 - x_*)\right\|^2}{2(L_h - \sigma_h)} \leq (1 - \frac{1}{\sqrt{L_h/\sigma_h} + 1})LD_h(x_*, x_0).$$

Upper bound can be achieved when $\nabla h(x_1) - \nabla h(x_*) = \sqrt{L_h \sigma_h}(x_1 - x_*)$.

$$f_1 - f_* \leq (1 - \frac{1}{\sqrt{L_h/\sigma_h} + 1})LD_h(x_*, x_0).$$

Numerically verified! Worst case (right hand side) can be reached.

Case 1: h - SC & Smooth

Further discussion:

- how to extend to N iterations? (restart schema? how?)
- or directly find dual solutions for N iterations? (tedious and difficult!)

Table of Contents

PAME

$$f^{t+1} = f^t + \eta \log\left(\frac{a}{r\left(\sum_k \zeta^k(f^t, g^t, \lambda^t)\right)}\right), \tag{63a}$$

$$g^{t+1} = g^t + \eta \log\left(\frac{b}{c\left(\sum_k \zeta^k(f^{t+1}, g^t, \lambda^t)\right)}\right), \tag{63b}$$

$$y^{t+1} = \mathrm{Proj}_{\Delta^N}\left(\lambda^t + (1-\theta)(\lambda^t - \lambda^{t-1})\right), \tag{63c}$$

$$\lambda^{t+1} = \mathrm{Proj}_{\Delta^N}\left(y^{t+1} + \tau \nabla_\lambda F(f^{t+1}, g^{t+1}, y^{t+1})\right). \tag{63d}$$

- flexible θ_t for each iteration?
- different anchoring point and approximation point?

Bibliography I

Bauschke, H. H., Bolte, J., and Teboulle, M. (2017).
A descent lemma beyond lipschitz gradient continuity: First-order methods revisited and applications.
Math. Oper. Res., 42:330–348.