Convex Optimization Report 3

Yunlei Lu

Dec. 14, 2020

Abstract

In this report, fast gradient algorithm for the smoothed primal problem, proximal gradient and fast proximal gradient algorithm for the primal problem are implemented.

1 The Group Lasso Problem

$$\min \frac{1}{2}||AX-B||_F^2 + \mu||X||_{1,2} \tag{1}$$

where $A \in \mathbb{R}^{m\times n}$, $B \in \mathbb{R}^{m\times l}$, $X \in \mathbb{R}^{n\times l}$ and $\mu > 0$.

Define

$$f(X) = \frac{1}{2}||AX-B||_F^2 + \mu||X||_{1,2}$$

$$g(X) = \frac{1}{2}||AX-B||_F^2 = \frac{1}{2}tr[(AX-B)^T(AX-B)]$$

$$h(X) = \mu||X||_{1,2} = \mu\sum_{i=1}^{n}||x^i||_2$$

where x^i is the i th row vector of X.

$g(X)$ is differentiable with $\nabla g(X) = A^T(AX-B)$ and $h(X)$ is non-differentiable at $||x^i||_2 = 0$.

2 Smoothing

We can solve the smoothed problem so that the objective function is differentiable.

Define

$$f_\gamma(X) = g(X) + h_\gamma(X) \tag{2}$$

$$h_\gamma(X) = \sum_{i=1}^{n} h_\gamma^i(X) \tag{3}$$

where

$$h_\gamma^i(X) = \begin{cases} \dfrac{||x^i||_2^2}{2\gamma}, & ||x^i||_2 \le \gamma \\ ||x^i||_2 - \dfrac{1}{2}\gamma, & ||x^i||_2 \ge \gamma \end{cases} \tag{4}$$

It can be shown that,

$$\nabla h_\gamma^i(X) = \begin{cases} \dfrac{x^i}{\gamma}, & ||x^i||_2 \le \gamma \\ \dfrac{x^i}{||x^i||_2}, & ||x^i||_2 \ge \gamma \end{cases} \tag{5}$$

$$\nabla h_\gamma(X) = (\nabla h^1_\gamma(X), ..., \nabla h^n_\gamma(X))^T \tag{6}$$

Thus, $f_\gamma(X)$ is differentiale:

$$\nabla f_\gamma(X) = \nabla g(X) + \nabla h_\gamma(X) \tag{7}$$

For small smoothing parameter γ, the smoothed problem can be a good approximation of the original problem.

3 Subgradient Method

We can also obtain the subgraident at $||x^i||_2 = 0$ to solve the problem using subgradient method.
The subgradient of $h(X)$ can be given as

$$\partial h(X) = (\partial ||x^1||_2, ..., \partial ||x^n||_2)^T \tag{8}$$

where

$$\partial ||x^i||_2 = \begin{cases} \dfrac{x^i}{||x^i||_2}, & ||x^i||_2 \neq 0 \\ \{g|||g||_2 \leq 1\}, & ||x^i||_2 = 0 \end{cases} \tag{9}$$

Thus,

$$\partial f(X) = A^T(AX - B) + \partial h(X) \tag{10}$$

4 Proximal Gradient Method

For unconstrained convex optimization problem:

$$\min f(x) = g(x) + h^*(x)$$

where g is convex and differentiable, h is convex with inexpensive prox-operator, the proximal gradient algorithm can be implemented to solve the convex optimizaiton problem.

Proximal gradient algorithm:

$$x^{(k)} = prox_{\alpha_k h^*}\Big(x^{(k-1)} - \alpha_k \nabla g(x^{(k-1)})\Big) \tag{11}$$

where $\alpha_k > 0$ is the step size, which is constant, diminishing or given by line search.

For the group lasso problem, the proximal operator of $\mu h(X)$ can be given as:

$$prox_{\alpha\mu h}(X) = \operatorname*{argmin}_U \Big(\alpha\mu ||U||_{1,2} + \frac{1}{2}||U - X||^2_F\Big) \tag{12}$$

$$= X - \alpha\mu \; \partial h(X) \tag{13}$$

5 Fast Gradient Method (FISTA)

For the same problem mentioned in *Proximal Gradient Method*,

Fast Iterative Shrinkage-Thresholding Algorithm (FISTA):
Choose $x^{(-1)} = x^{(0)} = v^{(0)}$, for $k \geq 1$:

$$y = (1-\theta_k)x^{(k-1)} + \theta_k v^{(k-1)} \tag{14}$$

$$x^{(k)} = prox_{\alpha_k h}\Bigg(y - \alpha_k \nabla g(y)\Bigg) \tag{15}$$

$$v^{(k)} = x^{(k-1)} + \frac{1}{\theta_k}(x^{(k)} - x^{(k-1)}) \tag{16}$$

where $\theta_k = \frac{2}{k+1}$.

For the smoothed problem, the proximal operator of $\mu h(X)$ can be given as:

$$\begin{aligned} prox_{\alpha\mu h_\gamma}(X) &= \underset{U}{\operatorname{argmin}}\Bigg(\alpha\mu h_\gamma(U) + \frac{1}{2}||U - X||_F^2\Bigg) && (17)\\ &= X - \alpha\mu \ \nabla h_\gamma(X) && (18)\end{aligned}$$

6 Algorithm

For the original optimization problem, suppose the given regularization parameter is μ. The strategy is to start from $\mu_1 > \mu$ to the given μ. The sub-problem with parameter μ_t is solved using FISTA or Proximal Gradient Method, then the parameter is updated by $\mu_{t+1} = \max\{\eta\mu_t, \mu\}$, where $\eta = 0.1$.

Algorithm 1 FISTA with fixed step size

```
Initialize $X^{(0)} = V^{(0)}$ and $\alpha = \lambda_{max}(A^T A)$
for $k = 1, 2, \dots, MaxItrInn$ do
    $\theta_k = \frac{1}{k+2}$
    $Y = (1-\theta_k)X^{(k-1)} + \theta_k V^{(k-1)}$
    $X^{(k)} = prox_{\alpha h}\Bigg(Y - \alpha\nabla g(Y)\Bigg)$
    $V^{(k)} = X^{(k-1)} + \frac{1}{\theta_k}(X^{(k)} - X^{(k-1)})$

    $f_{best}(k) = \min\left(f_{\mu_t}(X^{(k)}), f_{best}(k-1)\right)$

    if $k > 10$ and $\left|f_{best}(\max(k-10,1)) - f_{best}(k)\right| < 10^{-5-t}$ then
        break
    end if
    if $k == MaxItr$ then
        flag = 1
    else
        flag = 0
    end if
end for
```

Algorithm 2 Proximal Gradient Method

```
    $\alpha = \lambda_{max}(A^T A)$
    for $k = 1, 2, \dots, MaxItrInn$ do
        $X^{(k)} = prox_{\alpha h}\Big(X^{(k-1)} - \alpha \nabla g(X^{(k-1)})\Big)$
        $f_{best}(k) = \min\left(f_{\mu_t}(X^{(k)}), f_{best}(k-1)\right)$
        if $k > 10$ and $\big|f_{best}(\max(k-10, 1)) - f_{best}(k)\big| < 10^{-5-t}$ then
            break
        end if
        if $k == MaxItr$ then
            flag = 1
        else
            flag = 0
        end if
    end for
```

Algorithm 3 Optimization Algorithm

```
    Initialize $\mu_t = \mu_1$ and $t = 1$
    while $k < MaxItr$ do
        Update $X$ and $f_{best}$ using Algorithm 2 or Algorithm 3
        if $flag == 1$ then
            $\mu_{t+1} = \max(\eta\mu_t, \mu)$
            $t \leftarrow t + 1$
        end if
        if $\mu_t == \mu$ and $(|f_{best}(k) - f_{best}(k-1)| < 10^{-10}$ or $||\nabla g(X^{(k)})|| < 10^{-8})$ then
            break
        end if
    end while
```

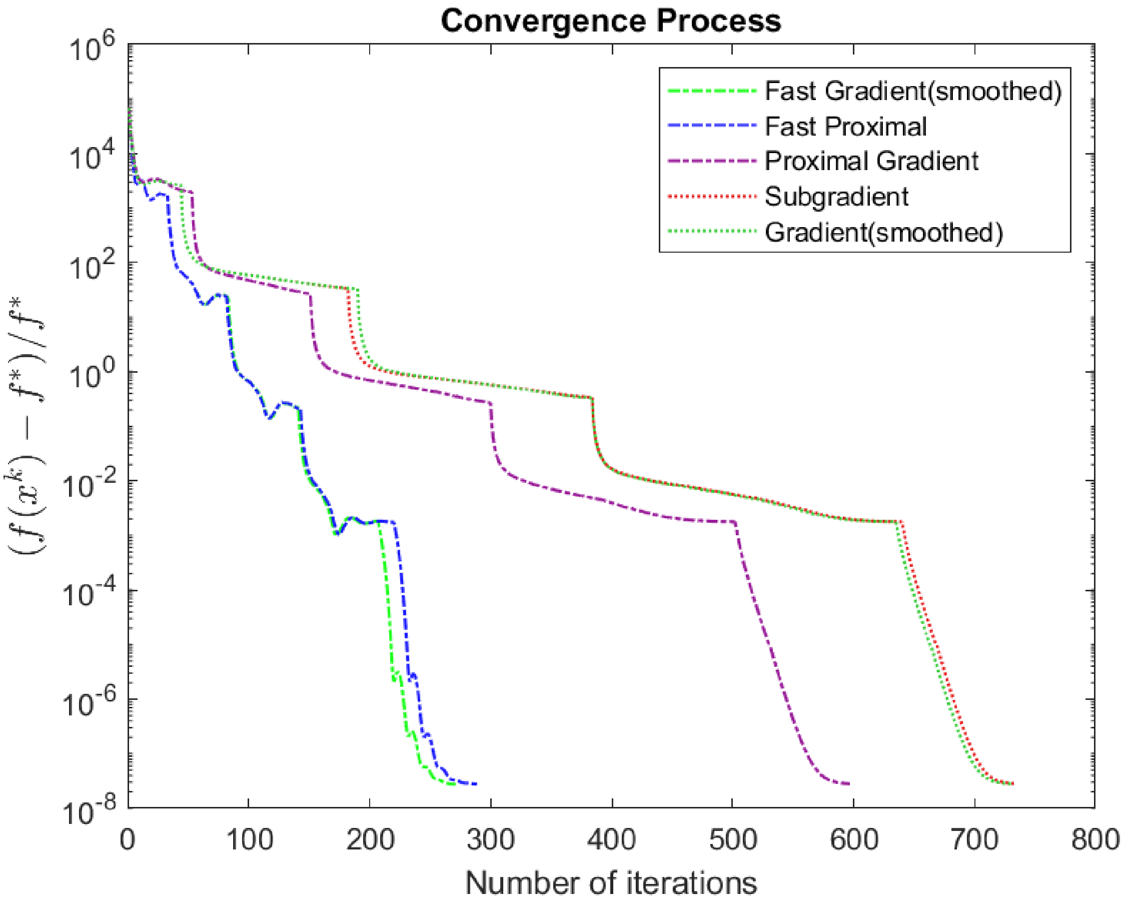

Figure 1: Convergence of Algorithms

7 Numerical Test Results

The test results are generated using data given in the file. The results are shown in Table 1 and Figure 1.

Solver	CPU	Iter	Optimal Value	Sparsity	Error-to-Exact
CVX-Mosek	1.028	-1	0.58055637	0.105	3.776 E-05
CVX-Gurobi	0.924	-1	0.58055623	0.103	3.746 E-05
Subgradient	1.380	733	0.58055625	0.100	3.691 E-05
Gradient(smoothed)	1.462	728	0.58055625	0.100	3.691 E-05
Fast Gradient(smoothed)	0.537	269	0.58055625	0.100	3.679 E-05
Proximal Gradient	1.277	595	0.58055625	0.100	3.691 E-05
Fast Proximal Gradient	0.695	288	0.58055625	0.100	3.678 E-05

Table 1: Test results

8 Summary and Conclusions

It can be observed from Figure 1 that the Proximal Gradient Descent Method converges faster than the Subgradient Method and Gradient Method for the smoothed problem, FISTA for both original and smoothed problem converges much faster than the Proximal Method only. The optimal value obtained by the algorithms above are almost the same, while the FISTA methods have a smaller error to exact solution than other algorithms. The sparsity if perfectly 0.1 for all the algorithms.