homework5第二次作业

1.proximal gradient method

- unconstrained optimization with objective splite in two components

$$minimize \;\; f(x) = g(x) + h(x)$$

  - g convex, differentiable,dom g = R^n
  - h convex with inexpensive pro-operator
  - proximal gradient algorithm

$$x^{(k)} = prox_{t_k h}(x^{(k-1)} - t_k \nabla g(x^{(k-1)}))$$

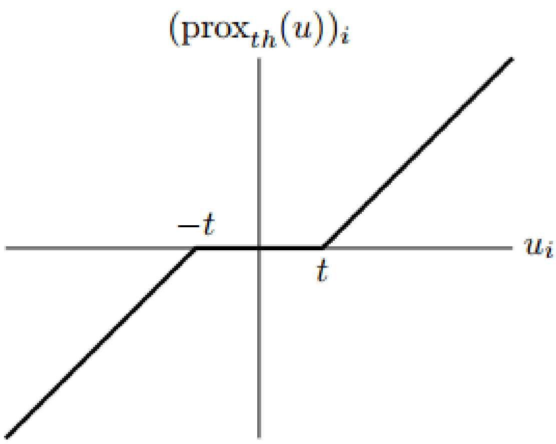
    - $t_k > 0$ is step size,constant or determined by line search
- 具体到本问题：
  - $min \;\; \frac{1}{2}||Ax-b||_2^2 + \mu||x||_1$
  - $x = x_0$

Soft-thresholding: special case with $h(x) = \|x\|_1$

$$x^+ = \mathrm{prox}_{th}\left(x - t\nabla g(x)\right)$$

where

  - $$(\mathrm{prox}_{th}(u))_i = \begin{cases} u_i - t & u_i \geq t \\ 0 & -t \leq u_i \leq t \\ u_i + t & u_i \leq -t \end{cases}$$



  - $t_k = 1/L$ ，$L = \lambda_{max}(A^T A)$
  - 终止条件：$x^* = prox_{th}(x^*)$

- 收敛速度：$O(\frac{1}{k})$
  - for fixed step size $t_k = 1/L$

    $$\|x^{(k)} - x^\star\|_2^2 \le c^k \|x^{(0)} - x^\star\|_2^2, \qquad c = 1 - \frac{m}{L}$$

    i.e., linear convergence if g is strongly convex ($m > 0$)
- $$f(x^{(k)}) - f^* \le \frac{1}{k}\sum_{i=1}^{k}(f(x^{(i)}) - f^\star) \le \frac{1}{2kt}\|x^{(0)} - x^\star\|_2^2$$
-
- 加速方法参考

  Another common situation in which a further efficiency improvement is possible is when the lasso problem is to be solved for many values of γ. For example, we might solve the problem for 50 values of γ, log spaced on the interval $[0.01\gamma_{\max}, \gamma_{\max}]$, where $\gamma_{\max} = \|A^T b\|_\infty$ is the critical value of γ above which the solution is $x^\star = 0$.

  A simple and effective method in this case is to compute the solutions in turn, starting with $\gamma = \gamma_{\max}$, and initializing the proximal gradient algorithm from the value of $x^\star$ found with the previous, slightly larger, value of γ. This general technique of starting an iterative algorithm from a solution of a nearby problem is called *warm starting*. The same idea works for other cases, such as when we add or delete rows and columns of A, corresponding to observing new training examples or measuring new features in a regression problem. Warm starting can thus permit the (accelerated) proximal gradient method to be used in an online or streaming setting.

2.accelerate proximal gradient method

- $$\begin{aligned} y &= x^{(k-1)} + \frac{\sqrt{t_k}}{\sqrt{t_{k-1}}}\frac{1-\sqrt{mt_{k-1}}}{1+\sqrt{mt_k}}(x^{(k-1)} - x^{(k-2)}) \\ x^{(k)} &= \mathrm{prox}_{t_k h}(y - t_k \nabla g(y)) \end{aligned}$$

- 版本很多，这里实现的是上面比较简单的方法。
- $t_k = 1/L$
- $L = \lambda_{max}(A^TA)$
- $m = \lambda_{min}(A^TA)$

- $$y = x^{(k-1)} + \frac{1-\sqrt{m/L}}{1+\sqrt{m/L}}(x^{(k-1)} - x^{(k-2)})$$

- 收敛速度

therefore,

$$f(x^{(k)}) - f^\star \le \frac{\theta_k^2}{2t}\|x^{(0)} - x^\star\|_2^2 = \frac{2L}{(k+1)^2}\|x^{(0)} - x^\star\|_2^2$$

3.gradient decent with smoothing method

- $$\phi_\mu(z) = \begin{cases} z^2/(2\mu) & |z| \le \mu \\ |z| - \mu/2 & |z| \ge \mu \end{cases}$$

trade-off in amount of smoothing (choice of μ)

-
  - large L_μ (less smoothing) gives more accurate approximation
  - small L_μ (more smoothing) gives faster convergence
- 这里$L_\mu = L + 1/\mu$
- $L = \lambda_{max}(A^T A)$
- efficiency in practice can be improved by decreasing μ gradually
- 复杂性：见下面的表格

4. fast gradient decent with smoothing method

- $y = x^{(k-1)} + \frac{k-2}{k+1}(x^{(k-1)} - x^{(k-2)})$
- $x^k = y - t_k \nabla g(y)$

5.收敛速度理论对比

first-order convex optimization methods	iterations
subgradient method	$O((G/\epsilon)^2)$
proximal gradient method	$O(L/\epsilon)$
fast proximal gradient method	$O(\sqrt{L/\epsilon})$
gradient method with smoothing	$O(L/\epsilon^2)$
fast gradient method with smoothing	$O(\sqrt{L}/\epsilon)$

- 其中L 为f的 Lipschitz constant
- ϵ-suboptimal point of $f(x)$

6.实际实验的收敛性