Homework 5 for "Convex Optimization" Part. 3

1500010611 汪祎非

We consider the $l1$-regularized problem

$$\min_{x} \frac{1}{2}\|Ax-b\|_2^2 + \mu\|x\|_1 \tag{1}$$

where $A \in \mathbb{R}^{m\times n}, b \in \mathbb{R}^m$ and $\mu > 0$ are given. We denote $f(x) = \frac{1}{2}\|Ax-b\|_2^2 + \mu\|x\|_1$, $g(x) = \frac{1}{2}\|Ax-b\|_2^2$, $h(x) = \|x\|_1$.

1 Problem 3(g)

The original problem 1 is equivalent to

$$\begin{cases} \min \ \frac{1}{2}\|y\|_2^2 + \mu\|x\|_1 \\ \text{s.t.} \quad Ax - b = y \end{cases} \tag{2}$$

where $x \in \mathbb{R}^n$, $y \in \mathbb{R}^m$.

We apply continuation strategy. We have three parameters α, M_1, M_2 for continuation strategy and one parameter M_3 for Newton method. We set $\mu_0 = \max\{\mu, \alpha\|A^T b\|_\infty\}$ and set $i = 0$. For each μ_i, problem 2 turns to be

$$\begin{cases} \min \ \frac{1}{2}\|y\|_2^2 + \mu_i\|x\|_1 \\ \text{s.t.} \quad Ax - b = y \end{cases} \tag{3}$$

The corresponding Lagrangian is

$$\begin{aligned} L(x, y, z) &= \frac{1}{2}\|y\|_2^2 + \mu_i\|x\|_1 + z^T(Ax - b - y) \\ &= -b^T z + g(y) - z^T y + \mu_i h(x) - (A^T z)^T x \end{aligned} \tag{4}$$

where $z \in \mathbb{R}^m$. Then, we have

$$\begin{aligned}
\min_{x,y} L(x,y,z) &= -b^T z + \min_y (g(y) - z^T y) + \min_x (\mu_i h(x) - (A^T z)^T x) \\
&= -b^T z - \max_y (z^T y - g(y)) + \min_x ((A^T z)^T x - \mu_i h(x)) \\
&= -b^T z - g^*(z) - \mu_i h^*(A^T z/\mu_i)
\end{aligned}$$

And we know that $g^*(z) = \frac{1}{2}\|z\|_2^2$, $h^*(z) = \begin{cases} 0 & \|z\|_\infty \le 1 \\ +\infty & \|z\|_\infty > 1 \end{cases}$. Therefore, the dual problem for problem 2 is

$$\begin{cases} \min \ \dfrac{1}{2}\|z\|_2^2 + b^T z \\ \text{s.t.} \quad \|A^T z\|_\infty \le \mu_i \end{cases} \tag{5}$$

It is equivalent to

$$\begin{cases} \min \ \dfrac{1}{2}\|z\|_2^2 + b^T z \\ \text{s.t.} \quad A^T z = w, \quad \|w\|_\infty \le \mu_i \end{cases} \tag{6}$$

where $w \in \mathbb{R}^n$.

The augmented Lagrangian for problem 6 is

$$L_t(z,w,\lambda) = \frac{1}{2}\|z\|_2^2 + b^T z + \lambda^T (A^T z - w) + \frac{t}{2}\|A^T z - w\|_2^2 \tag{7}$$

where $\lambda \in \mathbb{R}^n$ and t is a constant.

At first, we set $z^0 = 0, w^0, \lambda^0 = 0$. For given (z^k, w^k, λ^k), we consider the relation ship between w^{k+1} and z^{k+1} if w^{k+1} and z^{k+1} minimize $L_t(w, z, \lambda^k)$.

$$\begin{aligned}
w^{k+1} &= \arg \min_{\|w\|_\infty \le \mu_i} L_t(z^{k+1}, w, \lambda^k) \\
&= \arg \min_{\|w\|_\infty \le \mu_i} \frac{1}{2}\|z^{k+1}\|_2^2 + b^T z^{k+1} + (\lambda^k)^T (A^T z^{k+1} - w) + \frac{t}{2}\|A^T z^{k+1} - w\|_2^2
\end{aligned}$$

For each w_l, it is equivalent to minimize

$$-\lambda_l^k w_l + \frac{t}{2}(w_l - (A^T z^{k+1})_l)^2 = \frac{t}{2}\left((w_l)^2 - 2(\frac{\lambda_l^k}{t} + (A^T z^{k+1})_l) w_l + (A^T z^{k+1})_l^2\right)$$

If $|\lambda_l^k/t + (A^T z^{k+1})_l| \le \mu_i$, then the minimum of the above function is attained when $w_l = \frac{\lambda_l^k}{t} + (A^T z^{k+1})_l$. If $\lambda_l^k/t + (A^T z^{k+1})_l > \mu_i$, then the minimum of the above function is attained when $w_l = \mu_i$. And if $\lambda_l^k/t + (A^T z^{k+1})_l < -\mu_i$, then the minimum of the above function is attained when $w_l = -\mu_i$. Therefore, if we define the soft-thresholding function

$$S_{\mu_i}(w)_l = \begin{cases} 0 & |w_l| \le \mu_i \\ w_l - \mu_i & w_l > \mu_i \\ w_l + \mu_i & w_l < -\mu_i \end{cases} \tag{8}$$

Then, we get the relation ship between w^{k+1} and z^{k+1}

$$w^{k+1} = \lambda^k/t + A^T z^{k+1} - S_{\mu_i}(\lambda^k/t + A^T z^{k+1})$$

We now consider the following problem:

$$\arg\min_z \frac{1}{2}\|z\|_2^2 + b^T z + (\lambda^k)^T(\lambda_k/t - S_{\mu_i}(\lambda^k/t + A^T z)) + \frac{t}{2}\|\lambda_k/t - S_{\mu_i}(\lambda^k/t + A^T z)\|_2^2$$

It is equivalent to

$$\arg\min_z \frac{1}{2}\|z\|_2^2 + b^T z + \frac{t}{2}\|S_{\mu_i}(\lambda^k/t + A^T z)\|_2^2$$

We consider to use Newton's method to compute z^{k+1}. We start from $z^{(0)} = z^k$. We denote the derivative of the above equation by d_z and the Hessian of the above function at z by H_z. Suppose $v = \lambda^k/t + A^T z$ and A_l is the l-th column of A. Then, we have

$$d_z = z + b + t \sum_{|v_l| > \mu_i} A_l S_{\mu_i}(v)_l$$

$$H_z = I + t \sum_{|v_l| > \mu_i} A_l A_l^T$$

Nevertheless, calculation of the above formula is computationally costly. In practice, we use the following estimations of d_z and H_z:

$$d_z \approx \hat{d}_z = z + b + tAS_{\mu_i}(v)$$

$$H_z \approx \hat{H}_z = I + tAA^T$$

We iteratively compute

$$z^{(j+1)} = z^{(j)} - \hat{H}_{z^{(j)}}^{-1} \hat{d}_{z^{(j)}}$$

M_3 times. Suppose the outcome is denoted by $z^{k+1} = N_\mu(z^k, \lambda^k)$. Then, we get the update rule for $(z^{k+1}, w^{k+1}, \lambda^{k+1})$:

$$\begin{cases} z^{k+1} = N_\mu(z^k, \lambda^k) \\ w^{k+1} = \lambda^k/t + A^T z^{k+1} - S_{\mu_i}(\lambda^k/t + A^T z^{k+1}) \\ \lambda^{k+1} = \lambda^k + t(A^T z^{k+1} - w^{k+1}) \end{cases} \tag{ALM-upd}$$

If $\mu_i > \mu$, after M_1 iterations, we update $\mu_{i+1} = \max\{\mu, \alpha\mu_i\}$ and $i = i + 1$. we use z^k and λ^k as the initial value for z^0 and λ_0. From (ALM-upd), the value of w^0 is unnecessary.

If $\mu_i = \mu$, after M_2 iterations, we stop our algorithm. The algorithm of augmented Lagrangian method for the dual problem is given below.

Algorithm 1 Augmented Lagrangian method for the dual problem with continuation strategy

Input: t, continuation parameter α, M_1, M_2, Newton method parameter M_3.

```
Calculate μ_0 = max{α||A^T b||_∞, μ}. Let i = 0, z^0 = 0, λ^0 = 0, k = 0.
while μ_i > μ do
    while k < M_1 do
        Update (z^{k+1}, w^{k+1}, λ^{k+1}) by (ALM-upd), k = k + 1
    end while
    μ_{i+1} = max{μ, αμ_i}, i = i + 1
    Set z^0 = z^k, λ^0 = λ^k, k = 0.
end while
while k < M_2 do
    Update (z^{k+1}, w^{k+1}, λ^{k+1}) by (ALM-upd), k = k + 1
end while
return x = −λ_k
```

The Lagrangian for problem 6 is $L(z, w, \lambda) = \frac{1}{2}\|z\|_2^2 + b^T z + \lambda^T(A^T z - w)$. Then, we consider

$$\begin{aligned} &\min_{z\in\mathbb{R}^m, \|w\|_\infty \le \mu} \frac{1}{2}\|z\|_2^2 + b^T z + \lambda^T(A^T z - w) \\ =&\min_z(\frac{1}{2}\|z\|_2^2 + b^T z + \lambda^T A^T z) + \min_{\|w\|_\infty \le \mu} -\lambda^T w \\ =& -\frac{1}{2}\|A\lambda + b\|_2 - \mu\|\lambda\|_1 \end{aligned}$$

Therefore, $-\lambda$ from algorithm 1 is the approximation of the solution to the primal problem.

In practice, we take $t = 10^{-2}, \alpha = 0.1, M_1 = 10, M_2 = 10, M_3 = 1$.

2 Problem 3(h)

We also apply continuation strategy. We have three parameters α, M_1, M_2 for continuation strategy. We set $\mu_0 = \max\{\mu, \alpha\|A^T b\|_\infty\}$ and set $i = 0$. For each μ_i, based on the analysis from previous section, the dual problem is problem 6. And its corresponding augmented Lagrangian is 7.

For given (z^k, w^k, λ^k), we first update w^{k+1}. Based on the previous analysis, we can write its update rule as:

$$w^{k+1} = \lambda^k/t + A^T z^k - S_{\mu_i}(\lambda^k/t + A^T z^k)$$

Then, we update z^{k+1}. We consider

$$\begin{aligned}
&\arg\min_z \frac{1}{2}\|z\|_2^2 + b^T z + (\lambda^k)^T(A^T z - w^{k+1}) + \frac{t}{2}\|A^T z - w^{k+1}\|_2^2 \\
=&\arg\min_z \frac{1}{2} z^T(I + tAA^T)z + (b + A\lambda^k - tAw^{k+1})^T z \\
=&(I + tAA^T)^{-1}(-b - A\lambda^k + tAw^{k+1})
\end{aligned}$$

Therefore, the update rule of $(z^{k+1}, w^{k+1}, \lambda^{k+1})$ is given by:

$$\begin{cases}
w^{k+1} = \lambda^k/t + A^T z^k - S_{\mu_i}(\lambda^k/t + A^T z^k) \\
z^{k+1} = (I + tAA^T)^{-1}(-b - A\lambda^k + tAw^{k+1}) \\
\lambda^{k+1} = \lambda^k + t(A^T z^{k+1} - w^{k+1})
\end{cases} \tag{ADMM-d-upd}$$

If $\mu_i > \mu$, after M_1 iterations, we update $\mu_{i+1} = \max\{\mu, \alpha\mu_i\}$ and $i = i + 1$. we use z^k and λ^k as the initial value for z^0 and λ^0. From (ADMM-d-upd), the value of w^0 is unnecessary. If $\mu_i = \mu$, after M_2 iterations, we stop our algorithm. The algorithm of ADMM for the dual problem is given below.

Similarly, $-\lambda$ from algorithm 2 is the approximation of the solution to the primal problem.

In practice, we take $t = 10^{-2}, \alpha = 0.1, M_1 = 10, M_2 = 10$.

3 Problem 3(i)

We apply continuation strategy. We have three parameters α, M_1, M_2 for continuation strategy and one parameter s for linearization. We set $\mu_0 = \max\{\mu, \alpha\|A^T b\|_\infty\}$ and set $i = 0$. For each

Algorithm 2 ADMM for the dual problem with continuation strategy

Input: t, continuation parameter α, M_1, M_2.

Calculate $\mu_0 = \max\{\alpha \|A^T b\|_\infty, \mu\}$. Let $i = 0$, $z^0 = 0$, $\lambda^0 = 0$, $k = 0$.
while $\mu_i > \mu$ **do**
  while $k < M_1$ **do**
    Update $(z^{k+1}, w^{k+1}, \lambda^{k+1})$ by (ADMM-d-upd), $k = k + 1$
  end while
  $\mu_{i+1} = \max\{\mu, \alpha\mu_i\}$, $i = i + 1$
  Set $z^0 = z^k$, $\lambda^0 = \lambda^k$, $k = 0$.
end while
while $k < M_2$ **do**
  Update $(z^{k+1}, w^{k+1}, \lambda^{k+1})$ by (ADMM-d-upd), $k = k + 1$
end while
return $x = -\lambda_k$

μ_i, we consider another splitting form of the primal problem 1. It is equivalent to:

$$\begin{cases} \min \ \frac{1}{2}\|Ax - b\|_2^2 + \mu_i \|y\|_1 \\ \text{s.t.} \quad x = y \end{cases} \tag{9}$$

where $x \in \mathbb{R}^n$, $y \in \mathbb{R}^n$.

The corresponding augmented Lagrangian is

$$L_t(x, y, z) = \frac{1}{2}\|Ax - b\|_2^2 + \mu_i \|y\|_1 + z^T(x - y) + \frac{t}{2}\|x - y\|_2^2 \tag{10}$$

For given (x^k, y^k, z^k), we first update x^{k+1}. We have

$$\begin{aligned} x^{k+1} &= \arg\min_x L_t(x, y^k, z^k) = \arg\min_x \frac{1}{2}\|Ax - b\|_2^2 + \frac{t}{2}\|x - y^k + \frac{z^k}{t}\|_2^2 \\ &= (A^T A + tI)^{-1}(A^T b + t y^k - z^k) \end{aligned} \tag{11}$$

Although we have a closed form solution to $\arg\min_x L_t(x, y^k, z^k)$, the computation cost of calculating $(A^T A + tI)^{-1}(A^T b + t y^k - z^k)$ is a bit large. Therefore, we consider to use the following linear approximation to update x^{k+1}

$$x^{k+1} = x^k - s\left(A^T A x^k - A^T b + t(x^k - y^k + \frac{z^k}{t})\right) \tag{12}$$

where s is the step size of the linear approximation.

Then, we update y^{k+1} by:

$$
\begin{aligned}
y^{k+1} &= \arg\min_y L_t(x^{k+1}, y, z^k) = \arg\min_y \mu_i \|y\|_1 + \frac{t}{2}\|x^{k+1} - y + \frac{z^k}{t}\|_2^2 \\
&= \arg\min_y \frac{\mu_i}{t}\|y\|_1 + \frac{1}{2}\|y - (x^{k+1} + \frac{z^k}{t})\|_2^2 = S_{\mu_i/t}(x^{k+1} + \frac{z^k}{t})
\end{aligned}
$$

where $S_{\mu_i}(x)$ is the soft-thresholding function 8.

Finally, we update z^{k+1} by

$$z^{k+1} = z^k + t(x^{k+1} - y^{k+1})$$

In summary, the update rule of $(x^{k+1}, y^{k+1}, z^{k+1})$ is given by:

$$
\begin{cases}
x^{k+1} = x^k - s\left(A^T A x^k - A^T b + t(x^k - y^k + \frac{z^k}{t})\right) \\
y^{k+1} = S_{\mu_i/t}(x^{k+1} + \frac{z^k}{t}) \\
z^{k+1} = z^k + t(x^{k+1} - y^{k+1})
\end{cases}
\tag{ADMM-pl-upd}
$$

If $\mu_i > \mu$, after M_1 iterations, we update $\mu_{i+1} = \max\{\mu, \alpha\mu_i\}$ and $i = i + 1$. we use x^k, y^k and z^k as the initial value for x^0, y^0 and z^0.

If $\mu_i = \mu$, after M_2 iterations, we stop our algorithm. The algorithm of ADMM with linearization for the primal problem is given below.

Algorithm 3 ADMM with linearization for the primal problem with continuation strategy

Input: x^0, t, continuation parameter α, M_1, M_2, linearization step size s.

```
Calculate μ_0 = max{α||A^T b||_∞, μ}. Let i = 0, y^0 = x_0, z^0 = 0, k = 0.
while μ_i > μ do
    while k < M_1 do
        Update (x^{k+1}, y^{k+1}, z^{k+1}) by (ADMM-pl-upd), k = k + 1
    end while
    μ_{i+1} = max{μ, αμ_i}, i = i + 1
    Set x^0 = x^k, y^0 = y^k, z^0 = z^k, k = 0.
end while
while k < M_2 do
    Update (x^{k+1}, y^{k+1}, z^{k+1}) by (ADMM-pl-upd), k = k + 1
end while
return x = x_k
```

In practice, we take $t = 10^2, s = 5 \times 10^{-4}, \alpha = 0.1, M_1 = 200, M_2 = 200$. We also implement the ADMM for the primal problem with continuation strategy. The only difference is that we use 12 to update x^{k+1} instead of 11. The parameters for the ADMM for the primal problem with continuation strategy is: $t = 10^2, \alpha = 0.1, M_1 = 10, M_2 = 10$.

4 Numerical result

The whole test program is named *Test_hw05_03.m*. This time we run all algorithms mentioned before with same A and b. We compare the objective value of the primal problem. Suppose the value of objective function from cvx mosek is f_c, the value from proposed method is f_p, then the value of *objval to cvx mosek* is $\frac{f_p - f_c}{f_c}$. ALM is the augmented Lagrangian method for the dual problem. ADMM-d is the ADMM for the dual problem. ADMM-l-p is the ADMM with linearization for the primal problem. ADMM-p is the ADMM for the primal problem. We mark ADMM-p with * because this method is not required in the assignment. The numerical result is given in the following tables:

Table 1 Random seed is 2. The cpu time of cvx mosek is 1.06

Method	cpu time	objval to cvx mosek	error to cvx mosek
ALM	0.82	-1.87×10^{-6}	3.47×10^{-6}
ADMM-d	0.36	-8.69×10^{-7}	3.22×10^{-6}
ADMM-l-p	0.69	-1.87×10^{-6}	3.27×10^{-6}
ADMM-p*	1.22	-8.06×10^{-7}	3.17×10^{-6}

Table 2 Random seed is 7. The cpu time of cvx mosek is 1.27

Method	cpu time	objval to cvx mosek	error to cvx mosek
ALM	0.90	-3.09×10^{-6}	3.22×10^{-6}
ADMM-d	0.33	-1.93×10^{-6}	3.29×10^{-6}
ADMM-l-p	0.67	-3.09×10^{-6}	3.21×10^{-6}
ADMM-p*	1.31	-1.86×10^{-6}	3.26×10^{-6}

Table 3 Random seed is 9. The cpu time of cvx mosek is 1.05

Method	cpu time	objval to cvx mosek	error to cvx mosek
ALM	0.77	-2.06×10^{-6}	2.68×10^{-6}
ADMM-d	0.37	-6.32×10^{-7}	2.75×10^{-6}
ADMM-l-p	0.67	-2.06×10^{-6}	2.65×10^{-6}
ADMM-p*	1.23	-4.89×10^{-7}	2.76×10^{-6}