IE 8534 Fall 2020 – Homework 4

You must submit this homework to Canvas before 11:59 pm of Deember 16. You can scan your handwritten part and upload it along with your codes to Canvas. The instruction on coding is given below. No late submission will be accepted. **Substantial points will be deducted for those who copy/duplicate the others' work or provide work for copying/duplicating. Show all details of your work, not just the final answer.** If you have any question regarding homework marking, please contact the Teaching Assistant Chuan He (he000233@umn.edu).

Coding instruction: Codes can only be written in Matlab, and no other programming languages will be accepted. One should be able to execute all programs from matlab command prompt. Your code must be runnable otherwise no credit will be given. **Please specify instructions on how to run your program in the README file**. The submitted codes should be packed as a .zip or .rar file. A README file should be included for the explanation.

1. (**25 points**) Consider a matrix completion model

$$\min_{\|X\|_* \leq \vartheta} \underbrace{\frac{1}{2} \sum_{(i,j)\in\Omega} (X_{ij} - M_{ij})^2}_{g}, \tag{1}$$

where $\vartheta > 0$, $\|X\|_*$ is the nuclear norm of X, M is partially known matrix, and Ω is a set of a pair of indices.

Let $\vartheta = 12$ and M be a 20×10 matrix generated by the following Matlab codes:

```
rng('default');
M = zeros(20,10);
M(1:3,:) = rand(3,10);
r = rand(17,3);
for i = 1:17
    M(i+3,:) = (M(1,:)*r(i,1) + M(2,:)*r(i,2) + M(3,:)*r(i,3))/sum(r(i,:));
end
```

Let $\Omega = \{(I(i), J(i)) : 1 \leq i \leq 120\}$, where the index arrays I and J are generated by the following Matlab codes:

```
rng('default');
I = zeros(120,1); J = zeros(120,1);
r = randperm(200,120);
for k = 1:120
    i = ceil(r(k)/10); j = mod(r(k),10);
    if(j == 0); j = 10; end;
    I(k) = i; J(k) = j;
end
```

Implement the following methods for solving problem (1), starting with $X = 0$ and terminating once some X is found such that

$$\frac{\max\limits_{Y}\{\langle \nabla g(X), X - Y\rangle : \|Y\|_* \le \vartheta\}}{\max\{1, g(X)\}} \le 10^{-3}.$$

Check this termination condition once every 10 iterations to save computational cost. Report the final objective value, number of iterations, and CPU time.

(a) The proximal gradient method with fixed stepsize $t = 1/L$, where L is the Lipschitz constant of ∇g (See Slide 9-3).
(**Hint:** Let UDV^T be the singular value decomposition of Y, and d the vector extracted from the diagonal of D. Then the optimal solution of the problem

$$\min_{\|X\|_* \le \vartheta} \|X - Y\|_F^2$$

is given by UD^*V^T, where D^* is the diagonal matrix formed by aligning the vector d^* on its diagonal, and d^* is the optimal solution of the problem

$$\min_{\|x\|_1 \le \vartheta} \|x - d\|^2. \tag{2}$$

You can solve (2) by the Matlab codes provided in this link `https://stanford.edu/~jduchi/projects/DuchiShSiCh08/ProjectOntoL1Ball.m`)

(b) The conditional gradient method with step size $\tau_t = 2/(t + 2)$ (See Slide 11-5).
(**Hint:** Let u and v be the left and right singular vectors corresponding to the largest singular value $\sigma_{\max}$ of Y, which can be cheaply found by the Matlab built-in function ***svds*** (check the website `https://www.mathworks.com/help/matlab/ref/svds.html` for details). Then the optimal solution of the problem

$$\min_{\|X\|_* \le \vartheta} \langle Y, X\rangle$$

is given by $-\vartheta uv^T$, and its optimal value is $-\vartheta\sigma_{\max}$.)

2. (**40 points**)Consider the regression problem

$$\min_x \underbrace{\|Ax - b\| + \lambda\|x\|_1 + \mu \sum_{i=1}^{K} \|x_{\mathcal{G}_i}\| + \gamma \sum_{j=1}^{n-1} |x_j - x_{j+1}|}_{f(x)}, \tag{3}$$

for some $\lambda > 0$, $\mu > 0$ and $\gamma > 0$, where $\mathcal{G}_i \subset \{1, 2, \dots, n\}$ and $x_{\mathcal{G}_i}$ is a subvector of x indexed by $\mathcal{G}_i$ for $i = 1, \dots, K$. Suppose that X and y are generated by the following Matlab codes:

$$\begin{aligned} &rng('default');\\ &A = randn(500, 50);\\ &b = randn(500, 1);\end{aligned}$$

Let $\lambda = 0.01\|A^T b\|_\infty$, $\mu = 0.005\|A^T b\|_\infty$, $\gamma = 0.001\|A^T b\|_\infty$, $K = 5$, $n = 50$, and let $\{\mathcal{G}_i\}_{i=1}^5$ be the equal sequential partition of $\{1, 2, \ldots, 50\}$, that is, $\mathcal{G}_1 = \{1, \ldots, 10\}, \mathcal{G}_2 = \{11, \ldots, 20\}, \ldots, \mathcal{G}_{10} = \{41, \ldots, 50\}$. Apply the following methods to solve problem (3), starting with $x^{(1)} = 0$ and terminating them after running 2000 iterations. Report the smallest function value f_{best} found by each method over 2000 iterations, that is, $f_{\text{best}} = \min\{f(x^{(1)}), \ldots, f(x^{(2001)})\}$.

(a) The standard subgradient method with step size $\alpha_k = 1/k$ (see Slide 12-1).

(b) The standard subgradient method with the Polyak step size with $\gamma_k = 10/(10 + k)$ (see Slide 12-24).

(c) The filtered subgradient method with $\beta = 0.25$ and α_k given by the Polyak step size with $\gamma_k = 10/(10 + k)$ (see Slide 12-27).

(d) The CFM subgradient method with $\gamma_k = 1.5$ and α_k given by the Polyak step size with $\gamma_k = 10/(10 + k)$ (see Slide 12-27). (**Note:** The first γ_k is for the direction while the second one is for the step size.)

(e) The adaptive subgradient (AdaGrad) method with 0.01 (see Slide 15-15).

3. (**32 points**) Consider the classification problem

$$\begin{array}{ll} \min\limits_{w,b} & \overbrace{\mathbb{E}_\xi \left[\max(0, 1 - y_\xi(w^T X_\xi - b))\right] + \mu\|(w, b)\|}^{f(w,b)} \\ \text{s.t.} & \|(w, b)\| \le 1, \end{array} \tag{4}$$

where ξ is uniformly distributed in $\{1, \ldots, n\}$ for some positive integer n. Let $\mu = 10^{-2}$, $n = 800$, and $X = [X_1, \ldots, X_{800}]$ and y be generated by the following Matlab codes:

$$\begin{array}{l} rng('default'); \\ X = randn(100, 800); \\ y = ones(800, 1); \\ y(1 : 400) = -1; \end{array}$$

Set rng('default') and apply the following methods with $\omega(w, b) = (\|w\|^2 + b^2)/2$ to solve problem (4), starting with $(w_1, b_1) = (0, 0)$ and terminating them after running $30,000$ iterations. Report the smallest function value f_{best} found by each method over $30,000$ iterations. For parts (a)-(c),

$$f_{\text{best}} = \min\{f(w_1, b_1), \ldots, f(w_{30,001}, b_{30,001})\};$$

for part (d),

$$f_{\text{best}} = \min\{f(w_1^{av}, b_1^{av}), \ldots, f(w_{30,001}^{av}, b_{30,001}^{av})\};$$

and for part (e),

$$f_{\text{best}} = \min\{f(w_1^{ag}, b_1^{ag}), \ldots, f(w_{30,001}^{ag}, b_{30,001}^{ag})\}.$$

(a) The standard stochastic subgradient method with step size $\alpha_k = 1/k$ (see Slide13-3).

(b) The standard stochastic subgradient method with step size $\alpha_k = D/(G\sqrt{N})$ with $D = 1$, $G = \sqrt{\sum_{i=1}^n (\sqrt{\|X_i\|^2 + 1} + \mu)^2 / n}$, $N = 30,000$ (see Slide 13-3).

(c) The stochastic mirror descent method with

$$\gamma_k = \frac{1}{2\sqrt{(4M^2+\sigma^2)N}},$$

where

$$M = \mu + \frac{1}{n}\sum_{i=1}^n \sqrt{1+\|X_i\|^2}, \qquad \sigma = \sqrt{1+\frac{1}{n}\sum_{i=1}^n \|X_i\|^2}$$

(see Slide 16-7).

(d) The accelerated stochastic mirror descent method with $\beta_k = (k+1)/2$ and

$$\gamma_k = \frac{\sqrt{3}(k+1)}{2(4M^2+\sigma^2)^{1/2}(N+2)^{3/2}},$$

where M and σ are given above (see Slide 16-18).

4. (**15 points**) Consider a smooth nonconvex optimization problem

$$\min_{x\in\mathbb{R}^n} \; f(x) \tag{5}$$

with a finite optimal value f^*, where ∇f is Lipschitz continuous with parameter $L > 0$. Suppose that for any x, a stochastic gradient $G(x,\xi)$ is generated by a stochastic oracle that satisfies

$$\mathbb{E}_\xi[G(x,\xi)] = \nabla f(x), \qquad \mathbb{E}_\xi[\|G(x,\xi) - \nabla f(x)\|^2] \le \sigma^2.$$

Let $\{x^k\}$ be generated by the following stochastic gradient method:

$$x^{k+1} = x^k - \alpha_k G(x^k,\xi_k), \quad k = 1,2,\ldots$$

for some $\alpha_k \in (0, 1/L]$.

(a) Show that

$$\mathbb{E}_{\xi_k}[f(x^{k+1})] \le f(x^k) - \alpha_k\left(1 - \frac{L\alpha_k}{2}\right)\|\nabla f(x^k)\|^2 + \frac{L\sigma^2\alpha_k^2}{2}, \quad \forall k \ge 1.$$

(b) Show that

$$\mathbb{E}\left[\min_{1\le t\le k}\|\nabla f(x^t)\|^2\right] \le \frac{2(f(x^1)-f^*) + L\sigma^2\sum_{t=1}^k \alpha_t^2}{\sum_{t=1}^k \alpha_t}, \quad \forall k \ge 1.$$

5. (**18 points**) Consider a convex optimization problem

$$\min_{x\in\mathbb{R}^N} \; f(x), \tag{6}$$

where f is convex but possibly nonsmooth, and Lipschitz continuous with parameter $M > 0$. Let $\{x_1, x_2, \ldots, x_n\}$ be a partition of x and $\{\partial_1 f(x), \partial_2 f(x), \ldots, \partial_n f(x)\}$ the corresponding partition of $\partial f(x)$, where $x_i \in \mathbb{R}^{N_i}$ for $i = 1, \ldots, n$. Let us consider the following randomized block subgradient method for solving (6) with a given sequence of stepsizes $\{\alpha_k\}$, starting at $x^1 \in \mathbb{R}^N$.

```
For $k = 1, 2, \ldots$
    1) Choose $i_k \in \{1, \ldots, n\}$ uniformly at random;
    2) Compute $g_{i_k}^k \in \partial_{i_k} f(x^k)$;
    3) Set $x_{i_k}^{k+1} = x_{i_k}^k - \alpha_k g_{i_k}^k$, and $x_j^{k+1} = x_j^k$ for $j \neq i_k$.
End
```

Let x^* be an optimal solution of (6), $D = \|x^1 - x^*\|$, and let $\{x^k\}$ be generated above.

(a) Show that

$$\mathbb{E}_{i_k}\left[\frac{1}{2}\|x^{k+1} - x^*\|^2\right] \leq \frac{1}{2}\|x^k - x^*\|^2 + \frac{1}{n}\left[\alpha_k\left(f(x^*) - f(x^k)\right) + \frac{\alpha_k^2}{2}\|g^k\|^2\right], \; k = 1, 2, \ldots$$

for some $g^k \in \partial f(x^k)$.

(b) Show that

$$\mathbb{E}\left[\min_{1 \leq t \leq k} f(x^t)\right] - f^* \leq \frac{nD^2 + M^2 \sum_{t=1}^k \alpha_t^2}{2\sum_{t=1}^k \alpha_t}.$$

(c) Provide at least two choices of $\{\alpha_k\}$ so that the above method is convergent and justify your answer.

6. (**20 points**) Consider the graphical lasso model

$$\min_{X \succ 0} \underbrace{-\log\det(X) + \langle S, X\rangle + \lambda\sum_{i \neq j}|X_{ij}| + \frac{\mu}{2}\|X\|_F^2}_{f(X)} \tag{7}$$

for some $\lambda, \mu \geq 0$, where $X \succ 0$ means X is a symmetric positive definite matrix. Let S be a 100×100 positive definite matrix generated by the following Matlab codes:

```
rng('default');
A = randn(100, 100);  A = (A + A')/2;
r = randperm(10000, 4950);
for k = 1 : 4950
    i = ceil(r(k)/100);  j = mod(r(k), 100);
    if(j == 0);  j = 100;  end;
    A(i, j) = 0;  A(j, i) = 0;
end
A = A + (2 * norm(A)) * eye(100);
S = inv(A) + 0.1 * rand(100, 100);  S = (S + S')/2;
S = S − min(min(eig(S)) − 10^-4, 0) * eye(100);
```

Let $\lambda = 5 \times 10^{-2}$ and $\mu = 10^{-5}$. Apply ADMM method to solve model (7) with penalty parameter $t = 5$, 50, 500, respectively, starting with the initial Lagrangian multiplier $Z^{(0)} = 0$ and terminating when $f(X^{(k)}) - d(Z^{(k)}) \leq 10^{-3}$, where

$$d(Z) = \min_{X \succ 0, Y} \left\{ -\log\det(X) + \langle S, X \rangle + \lambda \sum_{i \neq j} |Y_{ij}| + \frac{\mu}{2} \|Y\|_F^2 + \langle Z, X - Y \rangle \right\}$$

Report the objective function value, number of iterations, CPU time, and comment on how the number of iterations changes as the penalty parameter t increases (see Slide 20-16).