Homework 11

Requirements:

1. Digital format (can be typeset or photos, ought to write clearly if written by hand), upload to `https://course.pku.edu.cn/`.
2. Submit by next class
3. A problem is not counted if nobody can work it out
4. Each homework 10 points; 1 point deducted for each week's delay

Problems:

1. [Projected gradient method.] Let $(\mathbf{v}, \mathbf{w})$ be the unique solution of

$$\begin{bmatrix} \mathbf{I} & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \mathbf{w} \end{bmatrix} = \begin{bmatrix} -\nabla f(\mathbf{x}) \\ \mathbf{0} \end{bmatrix}.$$

Show that

$$\mathbf{v} = \underset{\mathbf{A}\mathbf{u}=\mathbf{0}}{\operatorname{argmin}} \| -\nabla f(\mathbf{x}) - \mathbf{u}\|_2, \quad \mathbf{w} = \underset{\mathbf{y}}{\operatorname{argmin}} \|\nabla f(\mathbf{x}) + \mathbf{A}^T\mathbf{y}\|_2.$$

2. We consider the following problem

$$\min_{\mathbf{x}} f(\mathbf{x}) = \log\left(\sum_{i=1}^{n} e^{x_i}\right), \quad s.t. \quad \mathbf{A}\mathbf{x} = \mathbf{b}.$$

Use the following methods to solve it at $p = 100$ and $n = 500$ (you may generate $\mathbf{A}$ and $\mathbf{b}$ randomly, but make sure that the problem has a minimizer):

- Direct projected gradient, with inexact line search.
- Damped Newton's method with equality constraint.
- Dual approach.

Write a report to describe your settings and compare their performance ($\log(f(\mathbf{x}_k) - f^*)$ vs. iteration number, where f^* can be estimated by running the best algorithm many many iteations). Codes should also be handed in.

3. Consider the problem

$$\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{x}\|_2, \quad s.t. \quad \mathbf{A}\mathbf{x} = \mathbf{b}, \tag{1}$$

where $\mathbf{A} \in \mathbb{R}^{m\times n}, \mathbf{b} \in \mathbb{R}^m, m \le n$, and rank $\mathbf{A} = m$. Let $\mathbf{x}^*$ be the solution and $\mathbf{x}^*_\gamma = \underset{\mathbf{x}}{\operatorname{argmin}} \ \frac{1}{2}\|\mathbf{x}\|_2 + \gamma\|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2$.

Randomly generate $\mathbf{A} \in \mathbb{R}^{200\times 300}$ and $\mathbf{b} \in \mathbb{R}^{200}$, then solve problem (1) numerically by

a. the penalty method with the absolute value penalty function.

b. the penalty method with the Courant-Beltrami penalty function.

Hand in your code and report on parameter setting and the the comparison among them ($\log(f(\mathbf{x}_k) - f^*)$ vs. iteration number, where f^* has a closed-form solution). Note that which is faster really depends on your choice of parameters.