CS395T: Numerical Optimization for Graphics and AI: Homework III

1 Guideline

- Please complete **3** problems out of **8** problems, and please complete at least one problem in the theory session.
- You are welcome to complete more problems.

2 Programming

Each problem in this section counts as two.
Problem 1 and Problem 2. We are interested in solving the following low-rank matrix problem. Given a sparse observation pattern $G = (g_{ij})_{1 \le i,j \le n} \in \{0,1\}^{n \times n}$ and a data matrix $A = (a_{ij})_{1 \le i,j \le n} \in \mathbb{R}^{n \times n}$, our goal is to recover a low-rank matrix pair $B \in \mathbb{R}^{n \times r}, C \in \mathbb{R}^{r \times n}$ by minimizing

$$\underset{B,C}{\text{minimize}} \sum_{i=1}^{n} \sum_{j=1}^{n} g_{ij}(a_{ij} - (\boldsymbol{e}_i^T B)(C\boldsymbol{e}_j))^2 + \frac{\mu}{2}(\|B\|_{\mathcal{F}}^2 + \|C\|_{\mathcal{F}}^2),$$

where μ is a small constant, which makes the optimization problem non-degenerate. Implement a trust-region method for solving this problem, and compare it against alternating minimization of B and C.

3 Theory

Problem 3. Derive necessary and sufficient conditions for $\boldsymbol{p}^\star$ being a optimal solution to the following optimization problem:

$$\begin{aligned} \underset{\boldsymbol{p}}{\text{minimize}} \quad & \boldsymbol{g}^T \boldsymbol{p} + \frac{1}{2}\boldsymbol{p}^T B \boldsymbol{p} \\ \text{subject to} \quad & \boldsymbol{p}^T A \boldsymbol{p} \le d^2, \end{aligned} \tag{1}$$

where B is symmetric and A is positive semidefintie (not necessarily positive definite).

Problem 4. Derive a close-form solution for the 2D trust-region problem under the L1-norm:

$$\begin{aligned} \underset{x,y \in \mathbb{R}}{\text{minimize}} \quad & \begin{pmatrix} x & y \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} + \frac{1}{2} \begin{pmatrix} x & y \end{pmatrix} \begin{pmatrix} a_{11} & a_{12} \\ a_{12} & a_{22} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \\ \text{subject to} \quad & |x| + |y| \le d. \end{aligned} \tag{2}$$

Your solution should be as a close-form of $b_1, b_2, a_{11}, a_{12}, a_{22}, d$.

Problem 5. Derive an optimally condition for the following trust region sub-problem for optimizing rotations:

$$\begin{aligned} \underset{R\in SO(m)}{\text{minimize}} \quad & \text{Trace}(C^T R) \\ \text{subject to} \quad & \|R_0 - R\|_{\mathcal{F}}^2 \leq d^2. \end{aligned} \tag{3}$$

Here $R_0 \in SO(m)$ and $C \in \mathbb{R}^{m\times m}$ is a constant matrix.

Problem 6. We consider solving the following "sub-problem" using conjugate gradient descent:

$$\begin{aligned} \underset{\boldsymbol{p}}{\text{minimize}} \quad & \phi(\boldsymbol{p}) := \boldsymbol{g}^T\boldsymbol{p} + \frac{1}{2}\boldsymbol{p}^T B\boldsymbol{p} \\ \text{subject to} \quad & \boldsymbol{p}^T C\boldsymbol{p} \leq d^2. \end{aligned} \tag{4}$$

Here B is a real symmetric matrix and C is positive definite matrix.

The algorithm we consider is described as follows:

- **Step I:** Set $\boldsymbol{p}_0 = 0, \boldsymbol{r}_0 = -\boldsymbol{g}$. Solve $C\hat{\boldsymbol{r}_0} = \boldsymbol{r}_0$. Set $\boldsymbol{d}_0 = \hat{\boldsymbol{r}}_0, i = 0$.
- **Step II:** Compute $\gamma_i = \boldsymbol{d}_i^T B\boldsymbol{d}_i$. If $\gamma_i > 0$ then continue with Step III. Otherwise compute $\tau > 0$ so that $(\boldsymbol{p}_i + \tau\boldsymbol{d}_i)^T C(\boldsymbol{p}_i + \tau\boldsymbol{d}_i) = d^2$, set $\boldsymbol{p} = \boldsymbol{p}_i + \tau\boldsymbol{d}_i$ and terminate.
- **Step III:** Compute $\alpha_i = \frac{\boldsymbol{r}_i^T\hat{\boldsymbol{r}}_i}{\gamma_i}$, $\boldsymbol{p}_{i+1} = \boldsymbol{p}_i + \alpha_i\boldsymbol{d}_i$. If $\boldsymbol{p}_{i+1}^T C\boldsymbol{p}_{i+1} < d^2$ than Continue with step IV. Otherwise compute $\tau > 0$ so that $(\boldsymbol{p}_i + \tau\boldsymbol{d}_i)^T C(\boldsymbol{p}_i + \tau\boldsymbol{d}_i) = d^2$, set $\boldsymbol{p} = \boldsymbol{p}_i + \tau\boldsymbol{d}_i$ and terminate.
- **Step IV:** Compute $\boldsymbol{r}_{i+1} = \boldsymbol{r}_i - \alpha_i B\boldsymbol{d}_i$. If $\frac{\boldsymbol{r}_{i+1}^T C\boldsymbol{r}_{i+1}}{\boldsymbol{g}^T C\boldsymbol{g}} \leq \epsilon^2$ then set $\boldsymbol{p} = \boldsymbol{p}_{i+1}$ and terminate. Otherwise continue with Step V.
- **Step V:** Solve $C\hat{\boldsymbol{r}_{i+1}} = \boldsymbol{r}_{i+1}$. Compute $\beta_i = \frac{\boldsymbol{r}_{i+1}^T\hat{\boldsymbol{r}}_{i+1}}{\boldsymbol{r}_i^T\hat{\boldsymbol{r}}_i}$ and $\boldsymbol{d}_{i+1} = \hat{\boldsymbol{r}}_{i+1} + \beta_i\boldsymbol{d}_i$. Set $i := i + 1$ and continue with step II.

Let $\boldsymbol{p}_j, j = 0, \ddots, i$ be the iterates generated by the algorithm described above.

- Show that $\phi(\boldsymbol{p}_j)$ is strictly decreasing and $\phi(\boldsymbol{p}) \leq \phi(\boldsymbol{p}_i)$.
- Show that $\boldsymbol{p}_j^T C\boldsymbol{p}_j$ is strictly increasing for $j = 0, \ddots, i$ and $\boldsymbol{p}^T C\boldsymbol{p} \geq \boldsymbol{p}_j^T C\boldsymbol{p}_j$.