ECE/MATH 520, Spring 2008
Exam 2: Due Session 26

Solutions (version: April 29, 2008, 8:45) 75 mins.; Total 50 pts.

1. (16 pts.) The purpose of this question is to derive a recursive least-squares algorithm where we *remove* (instead of add) a data point. To formulate the algorithm, suppose we are given matrices $\boldsymbol{A}_0$ and $\boldsymbol{A}_1$ such that

$$\boldsymbol{A}_0 = \begin{bmatrix} \boldsymbol{A}_1 \\ \boldsymbol{a}_1^T \end{bmatrix},$$

where $\boldsymbol{a}_1 \in \mathbb{R}^n$. Similarly, suppose vectors $\boldsymbol{b}^{(0)}$ and $\boldsymbol{b}^{(1)}$ satisfy

$$\boldsymbol{b}^{(0)} = \begin{bmatrix} \boldsymbol{b}^{(1)} \\ b_1 \end{bmatrix},$$

where $b_1 \in \mathbb{R}$. Let $\boldsymbol{x}^{(0)}$ be the least-squares solution associated with $(\boldsymbol{A}_0, \boldsymbol{b}^{(0)})$, and $\boldsymbol{x}^{(1)}$ the least-squares solution associated with $(\boldsymbol{A}_1, \boldsymbol{b}^{(1)})$. Our goal is to write $\boldsymbol{x}^{(1)}$ in terms of $\boldsymbol{x}^{(0)}$ and the "removed" data point $(\boldsymbol{a}_1, b_1)$. As usual, let $\boldsymbol{G}_0$ and $\boldsymbol{G}_1$ be the Grammians associated with $\boldsymbol{x}^{(0)}$ and $\boldsymbol{x}^{(1)}$, respectively.

a. Write down expressions for the least-squares solutions $\boldsymbol{x}^{(0)}$ and $\boldsymbol{x}^{(1)}$ in terms of $\boldsymbol{A}_0$, $\boldsymbol{b}^{(0)}$, $\boldsymbol{A}_1$, and $\boldsymbol{b}^{(1)}$.

b. Derive a formula for $\boldsymbol{G}_1$ in terms of $\boldsymbol{G}_0$ and $\boldsymbol{a}_1$.

c. Let $\boldsymbol{P}_0 = \boldsymbol{G}_0^{-1}$ and $\boldsymbol{P}_1 = \boldsymbol{G}_1^{-1}$. Derive a formula for $\boldsymbol{P}_1$ in terms of $\boldsymbol{P}_0$ and $\boldsymbol{a}_1$. (The formula must not contain any matrix inversions.)

d. Derive a formula for $\boldsymbol{A}_0^T \boldsymbol{b}^{(0)}$ in terms of $\boldsymbol{G}_1$, $\boldsymbol{x}^{(0)}$, and $\boldsymbol{a}_1$.

e. Finally, derive a formula for $\boldsymbol{x}^{(1)}$ in terms of $\boldsymbol{x}^{(0)}$, $\boldsymbol{P}_1$, $\boldsymbol{a}_1$, and b_1. Use this and part c to write a recursive algorithm associated with successive removals of rows from $(\boldsymbol{A}_k, \boldsymbol{b}^{(k)})$.

Ans.: a. We have

$$\boldsymbol{x}^{(0)} = (\boldsymbol{A}_0^T \boldsymbol{A}_0)^{-1} \boldsymbol{A}_0^T \boldsymbol{b}^{(0)} = \boldsymbol{G}_0^{-1} \boldsymbol{A}_0^T \boldsymbol{b}^{(0)}.$$

Similarly,

$$\boldsymbol{x}^{(1)} = (\boldsymbol{A}_1^T \boldsymbol{A}_1)^{-1} \boldsymbol{A}_1^T \boldsymbol{b}^{(1)} = \boldsymbol{G}_1^{-1} \boldsymbol{A}_1^T \boldsymbol{b}^{(1)}.$$

b. Now,

$$\begin{aligned} \boldsymbol{G}_0 &= \begin{bmatrix} \boldsymbol{A}_1^T & \boldsymbol{a}_1 \end{bmatrix} \begin{bmatrix} \boldsymbol{A}_1 \\ \boldsymbol{a}_1^T \end{bmatrix} \\ &= \boldsymbol{A}_1^T \boldsymbol{A}_1 + \boldsymbol{a}_1 \boldsymbol{a}_1^T \\ &= \boldsymbol{G}_1 + \boldsymbol{a}_1 \boldsymbol{a}_1^T. \end{aligned}$$

Hence,

$$\boldsymbol{G}_1 = \boldsymbol{G}_0 - \boldsymbol{a}_1 \boldsymbol{a}_1^T.$$

c. Using the Sherman-Morrison formula,

$$\begin{aligned}
\boldsymbol{P}_1 &= \boldsymbol{G}_1^{-1} \\
&= (\boldsymbol{G}_0 - \boldsymbol{a}_1\boldsymbol{a}_1^T)^{-1} \\
&= \boldsymbol{G}_0^{-1} - \frac{\boldsymbol{G}_0^{-1}(-\boldsymbol{a}_1)\boldsymbol{a}_1^T\boldsymbol{G}_0^{-1}}{1+(-\boldsymbol{a}_1)^T\boldsymbol{G}_0^{-1}\boldsymbol{a}_1} \\
&= \boldsymbol{P}_0 + \frac{\boldsymbol{P}_0\boldsymbol{a}_1\boldsymbol{a}_1^T\boldsymbol{P}_0}{1-\boldsymbol{a}_1^T\boldsymbol{P}_0\boldsymbol{a}_1}.
\end{aligned}$$

d. We have

$$\begin{aligned}
\boldsymbol{A}_0^T\boldsymbol{b}^{(0)} &= \boldsymbol{G}_0\boldsymbol{G}_0^{-1}\boldsymbol{A}_0^T\boldsymbol{b}^{(0)} \\
&= \boldsymbol{G}_0\boldsymbol{x}^{(0)} \\
&= (\boldsymbol{G}_1 + \boldsymbol{a}_1\boldsymbol{a}_1^T)\boldsymbol{x}^{(0)} \\
&= \boldsymbol{G}_1\boldsymbol{x}^{(0)} + \boldsymbol{a}_1\boldsymbol{a}_1^T\boldsymbol{x}^{(0)}.
\end{aligned}$$

e. Finally,

$$\begin{aligned}
\boldsymbol{x}^{(1)} &= \boldsymbol{G}_1^{-1}\boldsymbol{A}_1^T\boldsymbol{b}^{(1)} \\
&= \boldsymbol{G}_1^{-1}(\boldsymbol{A}_1^T\boldsymbol{b}^{(1)} + \boldsymbol{a}_1 b_1 - \boldsymbol{a}_1 b_1) \\
&= \boldsymbol{G}_1^{-1}\left(\boldsymbol{A}_0^T\boldsymbol{b}^{(0)} - \boldsymbol{a}_1 b_1\right) \\
&= \boldsymbol{G}_1^{-1}\left(\boldsymbol{G}_1\boldsymbol{x}^{(0)} + \boldsymbol{a}_1\boldsymbol{a}_1^T\boldsymbol{x}^{(0)} - \boldsymbol{a}_1 b_1\right) \\
&= \boldsymbol{x}^{(0)} - \boldsymbol{G}_1^{-1}\boldsymbol{a}_1\left(b_1 - \boldsymbol{a}_1^T\boldsymbol{x}^{(0)}\right) \\
&= \boldsymbol{x}^{(0)} - \boldsymbol{P}_1\boldsymbol{a}_1\left(b_1 - \boldsymbol{a}_1^T\boldsymbol{x}^{(0)}\right).
\end{aligned}$$

The general RLS algorithm for removals of rows is:

$$\begin{aligned}
\boldsymbol{P}^{(k+1)} &= \boldsymbol{P}_k + \frac{\boldsymbol{P}_k\boldsymbol{a}_{k+1}\boldsymbol{a}_{k+1}^T\boldsymbol{P}_k}{1-\boldsymbol{a}_{k+1}^T\boldsymbol{P}_k\boldsymbol{a}_{k+1}} \\
\boldsymbol{x}^{(k+1)} &= \boldsymbol{x}^{(k)} - \boldsymbol{P}_{k+1}\boldsymbol{a}_{k+1}\left(b_{k+1} - \boldsymbol{a}_{k+1}^T\boldsymbol{x}^{(k)}\right).
\end{aligned}$$

2. (10 pts.) Use the penalty method to solve the following problem analytically:

$$\begin{array}{ll}
\text{minimize} & x_1^2 + 2x_2^2 \\
\text{subject to} & x_1 + x_2 = 3.
\end{array}$$

Hint: Use the penalty function $P(x) = (x_1 + x_2 - 3)^2$. The solution you find must be exact, not approximate.

Ans.: First, we construct the unconstrained objective function with penalty parameter γ:

$$f(\boldsymbol{x}) = x_1^2 + 2x_2^2 + \gamma(x_1 + x_2 - 3)^2.$$

Because f is a quadratic with positive definite quadratic term, it is easy to find its minimizer:

$$\boldsymbol{x}_\gamma = \frac{1}{1+2/(3\gamma)} \begin{bmatrix} 2 \\ 1 \end{bmatrix}.$$

For example, we can obtain the above by solving the FONC:

$$\begin{aligned} 2(1+\gamma)x_1 + 2\gamma x_2 - 6\gamma &= 0 \\ 2\gamma x_1 + 2(2+\gamma)x_2 - 6\gamma &= 0. \end{aligned}$$

Now letting $\gamma \to \infty$, we obtain

$$\boldsymbol{x}^* = \begin{bmatrix} 2 \\ 1 \end{bmatrix}.$$

(It is easy to verify, using other means, that this is indeed the correct solution.)

3. (12 pts.) Consider a standard form LP problem. Suppose we start with an initial basic feasible solution $\boldsymbol{x}^{(0)}$ and we apply one iteration of the simplex algorithm to obtain $\boldsymbol{x}^{(1)}$.

As pointed out in class, it turns out that we can express $\boldsymbol{x}^{(1)}$ in terms of $\boldsymbol{x}^{(0)}$ as

$$\boldsymbol{x}^{(1)} = \boldsymbol{x}^{(0)} + \alpha_0 \boldsymbol{d}^{(0)},$$

where α_0 minimizes $\phi(\alpha) = f(\boldsymbol{x}^{(0)} + \alpha \boldsymbol{d}^{(0)})$ over all $\alpha > 0$ such that $\boldsymbol{x}^{(0)} + \alpha \boldsymbol{d}^{(0)}$ is feasible.

a. Show that $\boldsymbol{d}^{(0)} \in \mathcal{N}(\boldsymbol{A})$.

b. As usual, assume that the initial basis is the first m columns of $\boldsymbol{A}$, and the first iteration involves inserting $\boldsymbol{a}_q$ into the basis, where $q > m$. Let the qth column of the canonical augmented matrix be $\boldsymbol{y}_q = [y_{1q}, \ldots, y_{mq}]^T$.

   Express $\boldsymbol{d}^{(0)}$ in terms of $\boldsymbol{y}_q$.

c. Show that $\boldsymbol{d}^{(0)}$ is a descent direction if and only if $r_q < 0$.

Ans.: a. We have

$$\boldsymbol{A}\boldsymbol{d}^{(0)} = \boldsymbol{A}(\boldsymbol{x}^{(1)} - \boldsymbol{x}^{(0)})/\alpha_0 = (\boldsymbol{b} - \boldsymbol{b})/\alpha_0 = 0.$$

Hence, $\boldsymbol{d}^{(0)} \in \mathcal{N}(\boldsymbol{A})$.

b. From our discussion of moving from one BFS to an adjacent BFS, we deduce that

$$\boldsymbol{d}^{(0)} = \begin{bmatrix} -\boldsymbol{y}_q \\ \boldsymbol{e}_{q-m} \end{bmatrix}.$$

In other words, the first m components of $\boldsymbol{d}^{(0)}$ are $-y_{1q}, \ldots, -y_{mq}$, and all the other components are 0 except the qth component, which is 1.

c. Now, we know that $\boldsymbol{d}^{(0)}$ is a descent direction if and only if $\boldsymbol{c}^T\boldsymbol{d}^{(0)} < 0$. So it remains to show that $\boldsymbol{c}^T\boldsymbol{d}^{(0)} < 0$ if and only if $r_q < 0$. From part b, $\boldsymbol{c}^T\boldsymbol{d}^{(0)} = c_q - \sum_{i=1}^m c_i y_{iq} = r_q$, and the desired result follows.

4. (12 pts.) Suppose we are given a matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ and a vector $\boldsymbol{b} \in \mathbb{R}^m$ such that $\boldsymbol{b} \geq \boldsymbol{0}$. We are interested in an algorithm that, given this $\boldsymbol{A}$ and $\boldsymbol{b}$, is guaranteed to produce one of following two outputs: (1) If there exists $\boldsymbol{x}$ such that $\boldsymbol{Ax} \geq \boldsymbol{b}$, then the algorithm produces one such $\boldsymbol{x}$. (2) If no such $\boldsymbol{x}$ exists, then the algorithm produces an output to declare so.

Describe in detail how to design this algorithm based on the simplex method.

Ans.: First, we convert the inequality constraint $\boldsymbol{Ax} \geq \boldsymbol{b}$ into standard form. To do this, we introduce a variable $\boldsymbol{w} \in \mathbb{R}^m$ of surplus variables to convert the inequality constraint into the following equivalent constraint:

$$[\boldsymbol{A}, -\boldsymbol{I}] \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{w} \end{bmatrix} = \boldsymbol{b}, \quad \boldsymbol{w} \geq \boldsymbol{0}.$$

Next, we introduce variables $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^n$ to replace the free variable $\boldsymbol{x}$ by $\boldsymbol{u} - \boldsymbol{v}$. We then obtain the following equivalent constraint:

$$[\boldsymbol{A}, -\boldsymbol{A}, -\boldsymbol{I}] \begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{v} \\ \boldsymbol{w} \end{bmatrix} = \boldsymbol{b}, \quad \boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w} \geq \boldsymbol{0}.$$

This form of the constraint is now in standard form. So we can now use Phase I from the simplex method to implement an algorithm to find a vectors $\boldsymbol{u}$, $\boldsymbol{v}$, and $\boldsymbol{w}$ satisfying the above constraint, if one exists, or to declare that none exists. If one exists, we output $\boldsymbol{x} = \boldsymbol{u} - \boldsymbol{v}$; otherwise, we declare that no $\boldsymbol{x}$ exists such that $\boldsymbol{Ax} \geq \boldsymbol{b}$. By construction, this algorithm is guaranteed to behave in the way specified by the question.