Please provide complete and well-written solutions to the following exercises.

Due October 25, in the discussion section.

Homework 4

Exercise 1. Let A be a real $m \times n$ matrix. Let $x \in \mathbf{R}^n$ and let $b \in \mathbf{R}^m$. Show that the function $f \colon \mathbf{R}^n \to \mathbf{R}$ defined by $f(x) = \frac{1}{2}\left|\left|Ax - b\right|\right|^2$ is convex. Moreover, show that

$$\nabla f(x) = A^T(Ax - b), \qquad D^2 f(x) = A^T A.$$

Exercise 2. Let A be an $m \times n$ real matrix with $m \geq n$. Then A has rank n if and only if $A^T A$ is positive definite.

(Hint: $A^T A$ is automatically positive semidefinite by a previous exercise.)

Exercise 3. Show the following identity. Let A be an $r \times r$ real matrix, let U be an $r \times s$ real matrix, and let V be an $s \times r$ real matrix. Assume that A is invertible and that $I + VA^{-1}U$ is invertible, where I is the $s \times s$ identity matrix. Then $A + UV$ is invertible and

$$(A + UV)^{-1} = A^{-1} - (A^{-1}U)(I + VA^{-1}U)^{-1}(VA^{-1}).$$

In particular, if $s = 1$, we get the Sherman-Morrison formula:

$$(A + UV)^{-1} = A^{-1} - \frac{A^{-1}UVA^{-1}}{1 + VA^{-1}U}.$$

Exercise 4. Give a bound for the number of arithmetic operations needed in Algorithm **??**. Assume that P_n is known, so that computing P_n does not require any arithmetic operations. (Hint: your bound should be something like $(m - n)n^2$.) Compare this bound to simply minimizing $\left|\left|Ax - b\right|\right|^2$ directly. (In that case, you should need about mn^2 arithmetic operations.)

The key point here is that, once P_n is known, recursive least squares is much better. For example, if $m - n = 10$, then recursive least squares requires around $10n^2$ arithmetic operations. But minimizing $\left|\left|Ax - b\right|\right|^2$ directly would require mn^2 operations, which is much larger.

Exercise 5. Let $x^{(1)}, \ldots, x^{(m)}$ be vectors in $\mathbf{R}^2$ such that $\left|\left|x^{(i)}\right|\right| = 1$ for all $1 \leq i \leq m$. Show that there exist $i, j \in \{1, \ldots, m\}$ with $i \neq j$ such that $\langle x^{(i)}, x^{(j)} \rangle > 1 - 100/m^2$.

Exercise 6 (Logistic Regression)**.** Let $x^{(1)}, \ldots, x^{(m)} \in \mathbf{R}^n$ and let $y_1, \ldots, y_m \in \{0, 1\}$. For the sake of intuition, we can think of each vector $x^{(i)}$ as a vector of words in an email, and y_i classifies email $i \in \{1, \ldots, m\}$ as either spam ($y_i = 1$) or not spam ($y_i = 0$). Given this data, we would like to find a way to classify future emails as spam or not spam. (This is what a spam filter does.) For any $t \in \mathbf{R}$, define the logistic function $g \colon \mathbf{R} \to (0, 1)$ by

$$g(t) = \frac{1}{1 + e^{-t}}.$$

The function g is meant to be a differentiable approximation to a function whose output is either 0 or 1.

First, verify that $g'(t) = g(t)(1 - g(t))$ for any $t \in \mathbf{R}$. Then, consider the log-likelihood function $L\colon \mathbf{R}^n \to \mathbf{R}$ defined by

$$L(z) := \log\left(\prod_{i=1}^{m}[g(\langle z, x^{(i)}\rangle)]^{y_i}[1 - g(\langle z, x^{(i)}\rangle)]^{1-y_i}\right), \qquad \forall\, z \in \mathbf{R}^n.$$

We would like to maximize L. The idea here is that if L is large, then z is a set of parameters (or "weights") that accurately classifies known emails as spam or not spam. So, once we find z, and if we have some new email $x \in \mathbf{R}^n$, then $g(\langle z, x^{(i)}\rangle) \approx 0$ means the new email is probably not spam, and $g(\langle z, x^{(i)}\rangle) \approx 1$ means the new email is probably spam.

Show that

$$\nabla L(z) = \sum_{i=1}^{m}(y_i - g(\langle z, x^{(i)}\rangle))x^{(i)}, \qquad \forall\, z \in \mathbf{R}^n.$$

This computation then gives the formula for a gradient ascent method for maximizing L.

Exercise 7. Define

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \\ 1 & 3 \\ 2 & 3 \end{pmatrix}, \qquad b = \begin{pmatrix} 1 \\ 3 \\ 4 \\ 6 \end{pmatrix}.$$

Minimize the function $||Ax - b||^2$ over all $x \in \mathbf{R}^2$, either by hand, or using a computer program that you write yourself. In either case, use the recursive least squares method.

Verify that the x you found does actually minimize $||Ax - b||^2$.

Exercise 8. Let $c, x \in \mathbf{R}^n$, let $b \in \mathbf{R}^m$, and let A be an $m \times n$ real matrix.

It is possible to essentially put all of the variables of a linear program into the constraint. Show that the following linear program is equivalent to the standard one. (That is, the minimum/maximum values and the x achieving the minimum/maximum value is the same for both linear programs.)

$$\text{maximize} \quad t \quad \text{subject to the constraints}$$
$$\{t \in \mathbf{R}\colon t \leq c^T x,\ \forall\, x \in \mathbf{R}^n \text{ such that } Ax = b,\ x \geq 0\}.$$

Exercise 9. Let $c, x \in \mathbf{R}^n$, let $b \in \mathbf{R}^m$, and let A be an $m \times n$ real matrix. Show that the linear program

$$\text{minimize} \quad c^T x \quad \text{subject to the constraints}$$
$$Ax \leq b.$$

is equivalent to the following linear program in standard form

$$\text{minimize} \quad \begin{pmatrix} c \\ -c \\ 0 \end{pmatrix}^T \begin{pmatrix} x^+ \\ x^- \\ z \end{pmatrix} \quad \text{subject to the constraints}$$

$$\begin{pmatrix} A & -A & I \end{pmatrix} \begin{pmatrix} x^+ \\ x^- \\ z \end{pmatrix} = b, \qquad \begin{pmatrix} x^+ \\ x^- \\ z \end{pmatrix} \geq 0.$$

By equivalent, we mean that if $x \in \mathbf{R}^n$, and if we define $x^+ := \max(x, 0)$, $x^- := \max(-x, 0)$, then $x = x^+ - x^-$. Similarly, if $x^+, x^- \geq 0$, then we define $x = x^+ - x^-$. And the minimum values and the x achieving the minimum value for both linear programs is the same. (Here the maximum is defined component-wise, e.g. if $x = (-1, 2, 3)$, then $\max(x, 0) = (0, 2, 3)$.)