Foundations of Computational Math I Exam 1
In-class Exam
Open Notes, Textbook, Homework Solutions Only
Calculators Allowed
Monday 19 Octobert, 2009

Question	Points Possible	Points Awarded
1. Basics	25	
2. Complexity	25	
3. Finite Precision	20	
4. Least Squares	30	
Total Points	100	

Name:________________________________

Alias: ________________________________

to be used when posting anonymous grade list.

Problem 1

Each of the following questions has a brief straightforward answer and justification. You must provide both for full credit.

1.a. **(5 points)** Suppose you have k vectors $x_i \in \mathcal{S} \subseteq \mathbb{R}^n$, $1 \le i \le k$ where $\mathcal{S}$ is a subspace of $\mathbb{R}^n$. Give two conditions that, if shown, guarantee the set of vectors $(x_1,\ x_2, \ldots, x_k)$ is a basis for the subspace $\mathcal{S}$.

Solution:

Either of the pairs of conditions work:

- the x_i are linearly independent and $k = dim(\mathcal{S})$
- the x_i are linearly independent and $span\,[x_1, \ldots, x_k] = \mathcal{S}$

1.b. **(10 points)** Suppose $P \in \mathbb{R}^{n\times n}$ is a permutation matrix (not necessarily an elementary permutation). Is $P^{-1} = P^T$?

Solution: For a general permutation matrix we have

$$P^T = \begin{pmatrix} e_{i_1}^T \\ \vdots \\ e_{i_n}^T \end{pmatrix} \text{ and } P = \begin{pmatrix} e_{i_1} & \ldots & e_{i_n} \end{pmatrix}$$

where $(i_1, \ldots, i_n)$ is a permutation vector of the integers $1 \le i \le n$, and $P^T P = I_n$ follows from $e_i^T e_j = 0$ for $i \ne j$ and $e_i^T e_i = 1$.

Another short justification is as follows. P is simply the identity matrix with rows or columns moved around. The columns of P are a set of orthonormal vectors and which implies that P is an orthogonal matrix. Therefore, $P^T = P^{-1}$.

Finally, you could use an argument based on what we know about elementary permutations. If P_i is an elementary permutation then $P_i = P_i^T = P_i^{-1}$. Any general permutation P can be factored into elementarty permutations and the result follows from the laws of transposes and inverses:

$$\begin{aligned} P &= P_1 P_2 \cdots P_k \\ P^T &= P_k^T \cdots P_2 P_1^T = P_k \cdots P_2 P_1 \\ &= P_k^{-1} \cdots P_2^{-1} P_1^{-1} = (P_1 P_2 \cdots P_k)^{-1} \\ &= P^{-1} \end{aligned}$$

1.c. **(10 points)** Suppose $Q \in \mathbb{R}^{n\times n}$ is an orthogonal matrix. Is $\|Q\|_2 = 1$?

Solution: Using the definition of the matrix 2-norm and the fact that Q is an isometry yields:

$$\|Q\|_2 = \max_{\|x\|_2=1} \|Qx\|_2 = \max_{\|x\|_2=1} \|x\|_2 = \max_{\|x\|_2=1} 1 = 1$$

Problem 2

2.a

(10 points)

Consider computing the matrix vector product $y = Tx$, i.e., you are given T and x and you want to compute y. Suppose further that the matrix $T \in \mathbb{R}^{n\times n}$ is tridiagonal with constant values on each diagonal. For example, if $n = 6$ then

$$\begin{pmatrix} \alpha & \beta & 0 & 0 & 0 & 0 \\ \gamma & \alpha & \beta & 0 & 0 & 0 \\ 0 & \gamma & \alpha & \beta & 0 & 0 \\ 0 & 0 & \gamma & \alpha & \beta & 0 \\ 0 & 0 & 0 & \gamma & \alpha & \beta \\ 0 & 0 & 0 & 0 & \gamma & \alpha \end{pmatrix}$$

(i) Write a simple loop-based psuedo-code that computes $y = Tx$ for such a matrix $T \in \mathbb{R}^n$.

(ii) How many operations are required as a function of n?

(iii) How many storage locations are required as a function of n?

Solution:

$$\eta_i \leftarrow \gamma\xi_{i-1} + \alpha\xi_i + \beta\xi_{i+1}, \quad 2 \leq i \leq n-1$$

$$\eta_1 \leftarrow \alpha\xi_1 + \beta\xi_2$$

$$\eta_n \leftarrow \gamma\xi_{n-1} + \alpha\xi_n$$

The storage required is $2n+3$. The computational complexity is $5(n-2)+6 = 5n-4 = 5n + O(1)$.

2.b

(15 points)

For both LU factorization and Householder reflector-based orthogonal factorization, we have used elementary transformations, T_i, that can be characterized as rank-1 updates to the identity matrix, i.e.,

$$T_i = I + x_i y_i^T, \quad x_i \in \mathbb{R}^n \text{ and } y_i \in \mathbb{R}^n$$

Gauss transforms and Householder reflectors differ in the definitions of the vectors x_i and y_i. Maintaining computational efficiency in terms of a reasonable operation count usually implies careful application of associativity and distribution when combining matrices and vectors.

Suppose we are to evaluate

$$z = T_3 T_2 T_1 v = (I + x_3 y_3^T)(I + x_2 y_2^T)(I + x_1 y_1^T)v$$

where $v \in \mathbb{R}^n$ and $z \in \mathbb{R}^n$. Show that by using the properties of matrix-matrix multiplication and matrix-vector multiplication, the vector z can be evaluated in $O(n)$ computations (a good choice of version for an algorithm) or $O(n^3)$ computations (a very bad choice of version for an algorithm).

(i) Show that by using the properties of matrix-matrix multiplication and matrix-vector multiplication, the vector z can be evaluated in $O(n)$ computations (a good choice of version for an algorithm).

(ii) Show that by using the properties of matrix-matrix multiplication and matrix-vector multiplication, the vector z can be evaluated in $O(n^3)$ computations (a very bad choice of version for an algorithm).

Solution:

First we derive the bad choice:

$$\text{compute } T_1 = (I + x_1 y_1^T) \rightarrow O(n^2) \text{ operations}$$
$$\text{compute } T_2 = (I + x_2 y_2^T) \rightarrow O(n^2) \text{ operations}$$
$$\text{compute } T_3 = (I + x_3 y_3^T) \rightarrow O(n^2) \text{ operations}$$
$$\text{compute } T = (T_3(T_2 T_1)) \rightarrow O(n^3) \text{ operations from two matrix-matrix products}$$
$$\text{compute } z = Tv \rightarrow O(n^2) \text{ operations}$$
$$O(n^3) \text{ operations in total as desired}$$

Second we derive the best choice:

$$z = (I + x_3 y_3^T)(I + x_2 y_2^T)(I + x_1 y_1^T)v$$
$$\text{compute } v_1 = v + x_1(y_1^T v) \rightarrow O(n) \text{ operations from an inner product and a vector triad}$$
$$\text{compute } v_2 = v_1 + x_2(y_2^T v_1) \rightarrow O(n) \text{ operations from an inner product and a vector triad}$$
$$\text{compute } z = v_2 + x_3(y_3^T v_2) \rightarrow O(n) \text{ operations from an inner product and a vector triad}$$
$$O(n) \text{ operations in total as desired}$$

For completeness we note an $O(n^2)$ version also exists:

$$
\begin{gathered}
\text{compute } T_1 = (I + x_1 y_1^T) \rightarrow O(n^2) \text{ operations} \\
\text{compute } T_2 = (I + x_2 y_2^T) \rightarrow O(n^2) \text{ operations} \\
\text{compute } T_3 = (I + x_3 y_3^T) \rightarrow O(n^2) \text{ operations} \\
\text{compute } v_1 = T_1 v \rightarrow O(n^2) \text{ operations} \\
\text{compute } v_2 = T_2 v_1 \rightarrow O(n^2) \text{ operations} \\
\text{compute } z = T_3 v \rightarrow O(n^2) \text{ operations} \\
O(n^2) \text{ operations in total}
\end{gathered}
$$

Problem 3

Define the function $f(x) = x - 1$ on the domain $x > 1$. Let $x_0 \in \mathbb{R}$, $x_0 > 1$, and $x_1 = x_0(1+\delta)$ where $\delta \in \mathbb{R}$ with $|\delta| < 1$.

(3.a) **(10 points)** Determine the relative error between $f(x_1)$ and $f(x_0)$, and the relative condition number $\kappa_{rel}(x_0)$.

Solution: We have

$$\frac{|f(x_1) - f(x_0)|}{|f(x_0)|} = \frac{|x_0|}{|x_0 - 1|}|\delta|$$

$$\frac{|f(x_1) - f(x_0)|}{|f(x_0)|} \leq \kappa_{rel}(x_0)|\delta| \to \kappa_{rel}(x_0) = \frac{|x_0|}{|x_0 - 1|} = \frac{x_0}{x_0 - 1} \quad \text{since } x_0 > 1$$

This is consistent with the Taylor series form. Strictly speaking we should take

$$\kappa_{rel}(x_0) = \max\left(1, \frac{x_0}{x_0 - 1}\right)$$

(3.b) **(10 points)** Suppose $|\delta| < 10^{-7}$. Determine the region of values for x_0 for which the relative error between $f(x_1)$ and $f(x_0)$ is no more than 10^{-4}.

Solution:

We want

$$\frac{|f(x_1) - f(x_0)|}{|f(x_0)|} \leq 10^{-4}$$

and we are given that $|\delta| < 10^{-7}$. Therefore, since

$$\frac{|f(x_1) - f(x_0)|}{|f(x_0)|} \leq \kappa_{rel}|\delta|$$

we must have

$$\kappa_{rel} = \frac{x_0}{x_0 - 1} \leq 10^3$$

We therefore have

$$\frac{x_0}{x_0 - 1} = 1 + \frac{1}{x_0 - 1} \leq 10^3$$

$$\frac{1}{x_0 - 1} \leq 999$$

$$x_0 \geq 1 + \frac{1}{999}$$

So we can simply take $x_0 > 1.002$.

Problem 4

(30 points)

Recall we showed that the arithmetic mean of n real numbers $\eta_1, \ldots, \eta_n$ solves the least squares problem

$$\min_{\beta \in \mathbb{R}} \left\| \begin{pmatrix} \eta_1 \\ \eta_2 \\ \vdots \\ \eta_n \end{pmatrix} - \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} \beta \right\|_2$$

or using vector notation

$$\min_{\beta \in \mathbb{R}} \|y - e\beta\|_2$$

Solve the problem by transforming it with a single Householder reflector and verify that the solution is

$$\beta_{min} = \bar{\eta} = \frac{1}{n} \sum_{i=1}^{n} \eta_i$$

Note: $e^T e = \|e\|_2^2 = n$.

Solution:

Since this is a linear least squares problem with an $n \times 1$ coefficient matrix e, i.e., the vector of all 1's and a scalar for a solution we have

$$\min_{\beta \in \mathbb{R}} \left\| \begin{pmatrix} \eta_1 \\ \eta_2 \\ \vdots \\ \eta_n \end{pmatrix} - \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} \beta \right\|_2 = \min_{\beta \in \mathbb{R}} \|y - e\beta\|_2 = \min_{\beta \in \mathbb{R}} \|Hy - He\beta\|_2$$

where H is the Householder reflector defined such that $He = -\gamma e_1$. Therefore, we have

$$-\gamma \beta_{min} = \phi = e_1^T H y$$

We must therefore compute these parameters to show that β_{min} is the mean of η_i, $1 \leq i \leq n$.

We have

$$
\begin{gathered}
H = I + \alpha x x^T, \quad \gamma = \|e\|_2 = \sqrt{n} \\
x = e + \sqrt{n} e_1, \quad \|x\|_2^2 = x^T x = 2(n + \sqrt{n}) \\
e_1^T x = \xi_1 = 1 + \sqrt{n}, \quad x^T y = e^T y + \sqrt{n} e_1^T y = e^T y + \eta_1 \sqrt{n} \\
\alpha = -\frac{2}{x^T x} = -\frac{1}{n + \sqrt{n}}
\end{gathered}
$$

$$
\begin{aligned}
\phi = e_1^T (I + \alpha x x^T) y &= \eta_1 + \alpha \xi_1 x^T y \\
&= \eta_1 - \frac{x^T y \xi_1}{(n + \sqrt{n})} \\
&= \eta_1 - \frac{e^T y (1 + \sqrt{n}) + \eta_1 (n + \sqrt{n})}{(n + \sqrt{n})} \\
&= -e^T y \frac{(1 + \sqrt{n})}{(n + \sqrt{n})} \\
&= -\frac{e^T y}{\sqrt{n}}
\end{aligned}
$$

Therefore the 1×1 system we must solve yields

$$
\begin{aligned}
-\gamma \beta_{min} &= \phi \\
-\sqrt{n} \beta_{min} &= -\frac{e^T y}{\sqrt{n}} \\
\beta_{min} &= \frac{e^T y}{n} = \frac{1}{n} \sum_{i=1}^{n} \eta_i = \bar{\eta}
\end{aligned}
$$