CAAM 453: Numerical Analysis I

Homework Assignment 2: Conditioning and Stability; Floating Point Arithmetic.
Due: September 19, 2016

Note: Turn in all MATLAB code that you have written and turn in all output generated by your MATLAB functions/scripts. MATLAB functions/scripts must be commented, output must be formatted nicely, and plots must be labeled. You may freely use any MATLAB functions from this year’s CAAM 453 edition. If you do so, you do not need to include their code in your writeup unless you have modified them.

Problem 1: Conditioning (30 points)

Consider a lower triangular matrix $A$ in $\mathbb{R}^{n \times n}$ with unit diagonal elements, subdiagonal elements $A_{i+1,i} = \alpha$, $i = 1, \ldots, n - 1$, and all other entries zero. Here is an instance with $n = 5$:

\[
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
\alpha & 1 & 0 & 0 & 0 \\
0 & \alpha & 1 & 0 & 0 \\
0 & 0 & \alpha & 1 & 0 \\
0 & 0 & 0 & \alpha & 1
\end{bmatrix}
\]

(a) Compute the inverse of $A$.

(b) Compute the condition number of $A$ using the $\ell_\infty$-norm.

(c) Now you are trying to solve a linear system of the form $Ax = b$ with $b = [1, 1, \ldots, 1]^\top$. Suppose that you computed an approximate solution $x_{\text{approx}}$ to the correct answer $x_{\text{sol}} = A^{-1}b$. Then you computed $b_{\text{approx}} = Ax_{\text{approx}}$ and you obtained $b_{\text{approx}} = [1.0001, 1, \ldots, 1]^\top$.

Use the condition number of $A$ to give a bound on the relative error in $x_{\text{approx}}$ when $\alpha = 0.01$ and $\alpha = 100$. Work with the $\ell_\infty$-norm.

Problem 2: Counting floating point operations. Conditioning? (40 points)

Let us consider again the matrix $A$ of Problem 1.

(a) In class we counted the number of floating point operations required to compute the QR factorization of general rectangular or squared matrix. Can the order of that bound ($n^3$) be reduced for a squared matrix with only diagonal and subdiagonal elements like matrix $A$? First try the Householder reflectors and then the Givens rotations.

(b) Code $A$ and $x_{\text{sol}} = [1, 1, \ldots, 1]^\top$ in Matlab using the commands `diag` and `ones` for a general $n$ and $\alpha$. Set $n = 50$ and $\alpha = 2$. Also in Matlab compute $b = Ax_{\text{sol}}$. Now solve the system $Ax = b$ in two ways: (i) Using the QR factorization (function `qr`) and making use of only two instructions; (ii) Using the operator “\”. Compute absolute and relative errors in both cases and explain the results.
Problem 3: Backward error analysis for floating point arithmetic (30 points)

Let $|A|$ and $|x|$ denote the matrix and the vector formed by the absolute values of the components a matrix $A$ and a vector $x$, respectively. This notation will facilitates the calculations in this problem.

(a) Let $z$ and $w$ be two vectors in $\mathbb{R}^m$ and $\beta$ a real scalar. The purpose of the first part of this problem is to analyze the effect of computing $\beta z + w$ in floating point arithmetic.

Set $u = \beta z + w - f\ell(\beta z + w)$. Prove that

$$|u| \leq (2|\beta||z| + |w|) \varepsilon + O(\varepsilon^2),$$

where $\varepsilon \equiv \varepsilon_{\text{mach}}$ is the machine precision.

(b) From this show that

$$\|u\|_{\infty} \leq (2|\beta||z|_{\infty} + \|w\|_{\infty}) \varepsilon + O(\varepsilon^2),$$

where $\varepsilon \equiv \varepsilon_{\text{mach}}$ is the machine precision.

(c) Show that $f\ell(\beta z + w)$ can be written as $(I + D)^{-1}((I + E)\beta z + w)$ with $D$ and $E$ diagonal matrices with diagonal elements of the order of $\varepsilon$. How does this show that the computation of $f\ell(\beta z + w)$ is backward stable?

(d) **OPTIONAL:** Now we consider the effect of floating point arithmetic on matrix-vector multiplication and set $v = Ax + y - f\ell(Ax + y)$ where $A \in \mathbb{R}^{m \times n}$, $x \in \mathbb{R}^n$, and $y \in \mathbb{R}^m$. By induction on $n$ and using a), prove that

$$|v| \leq (n + 1) (|A||x| + |y|) \varepsilon + O(\varepsilon^2).$$