Indian Institute of Science, Bangalore
Department of Computational and Data Sciences (CDS)

DS284: Numerical Linear Algebra

Assignment 2 [Posted Sept 2, 2024]

Faculty Instructor: Dr. Phani Motamarri
TAs: Gourab Panigrahi, Srinibas Nandi, Nihar Shah, Rushikesh Pawar, Surya Neelakandan

Notations: Vectors and matrices are denoted below by bold faced lower case and upper case alphabets respectively.

Problem 1

Solution to this problem needs to be submitted by Sep 15 and will be graded

Recall in IEEE single precision binary floating point representation, we use 32 bits to represent numbers 1 bit is for sign, 8 bits for the exponent and 23 bits for the mantissa. Using normalized binary scientific notation a floating point number in IEEE single precision can be represented as

$$(-1)^s \times (1.f)_2 \times 2^{(exponent-127)}$$

Here $s = 0$ for positive numbers and $s = 1$ for negative numbers. f represents the bits in mantissa. Note the digit 1 in $1.f$ and is explicitly shown for clarity and all binary representations are normalized to take the form $1.f$. The subscript 2 in the above $1.f$ denotes that we are representing the digits in base 2.

Now, in this exercise we will construct a dummy floating point number system where we use 5 bits of precision to represent numbers. In this simplified floating point number system, let us assume that we are representing numbers such that the exponent field admits values -1, 0 and 1 only. Imagine only positive numbers are represented in this system. Then the normalized binary scientific notation in this toy system would be

$$(1.f)_2 \times 2^{(exponent-1)}$$

Note here we use a biased representation in the exponent field instead of using a separate sign bit for exponent. The value of this bias is 1. Recall our toy floating point system admits -1, 0, 1 in the exponent field. Thus a value of -1 in the exponent field means $exponent = 0$, value of 0 in exponent field means $exponent = 1$, value of 1 in exponent field means $exponent = 2$. In this normalized binary scientific notation, 3 bits are used to store f, 2 bits are used to store the $exponent$.

In this backdrop answer the following questions for the toy floating point system we constructed above:

(a) How many numbers can this toy system describe?

(b) Create a table with 3 columns. First column should contain normalized binary scientific notation of the form

$$(1.f)_2 \times 2^{(exponent-1)}$$

of all the above numbers. (Make sure the numbers you are representing here just use 5 bits to store them). Second column should contain the usual binary representation. Third column should contain decimal representation (base 10 representation). Arrange the numbers in increasing order in the base 10 representation.

(c) From the table above, what is the minimum real number and maximum real number you can represent using our toy floating point number system.

(d) What can you say about absolute gaps between the numbers? Are they constant or do they change with the magnitude of the number you are representing?

(e) What can you say about machine epsilon for our toy floating point system?

(Hint: Pick $\mathbf{x} \in \mathbb{R}$, there exists $\mathbf{x}' \in \mathbb{F}$ such that $\frac{|\mathbf{x}-\mathbf{x}'|}{|\mathbf{x}|} \leqslant \epsilon_{machine}$)

Problem 2

Let $\mathbf{A} \in \mathbb{R}^{n\times n}$ be invertible matrix and consider the solution of the problem $\mathbf{Ax} = \mathbf{b}$ for some given non-zero $\mathbf{b} \in \mathbb{R}^n$.

(a) Derive the relative condition number of the problem of computing $\mathbf{x}$ given $\mathbf{b}$ with respect to perturbations in $\mathbf{b}$.

(b) Find the value of the tight lower bound of the relative condition number obtained in (a).

Problem 3

Let $\mathbf{A} \in \mathbb{R}^{n\times n}$ be invertible matrix and suppose that $\mathbf{x}$ solves $\mathbf{Ax} = \mathbf{b}$ for some given non-zero $\mathbf{b}$. Consider perturbations $\Delta\mathbf{A} \in \mathbb{R}^{n\times n}$ of $\mathbf{A}$ satisfying the following in some given matrix norm induced by the vector norm $\|.\|$,

$$K(\mathbf{A})\|\Delta\mathbf{A}\| < \|\mathbf{A}\|$$

where $K(\mathbf{A})$ is the condition number of $\mathbf{A}$ in the given norm. Consider also some perturbation $\Delta\mathbf{b} \in \mathbb{R}^n$ of $\mathbf{b}$ and let $\mathbf{x} + \Delta\mathbf{x}$ solve

$$(\mathbf{A} + \Delta\mathbf{A})(\mathbf{x} + \Delta\mathbf{x}) = \mathbf{b} + \Delta\mathbf{b}$$

Prove that

$$\frac{\|\Delta \mathbf{x}\|}{\|\mathbf{x}\|} \leqslant \left[\frac{K(\mathbf{A})}{1 - K(\mathbf{A}) \frac{\|\Delta \mathbf{A}\|}{\|\mathbf{A}\|}} \left[\frac{\|\Delta \mathbf{A}\|}{\|\mathbf{A}\|} + \frac{\|\Delta \mathbf{b}\|}{\|\mathbf{b}\|} \right] \right]$$

Problem 4

Recall the following from class:

(i) For all $x \in \mathbb{R}$ there exists $|\epsilon| \leqslant \epsilon_{machine}$ such that $fl(x) = x(1 + \epsilon)$, where $fl(x)$ denotes floating point representation of x.

(ii) For all $x, y \in \mathbb{F}$ there exists $|\epsilon| \leqslant \epsilon_{machine}$ such that $x \circledast y = (x * y)(1 + \epsilon)$ where $*$ denotes one of the operators $+, -, \times, \div$ and let $\circledast$ be its floating point analogue. Note $\mathbb{F}$ is a discrete subset of $\mathbb{R}$ which denote floating point representation of the real numbers.

Each of the following describes an algorithm implemented on a computer satisfying the properties (i) and (ii) described above. State with proper arguments whether the following algorithms are backward stable, stable but not backward stable, or unstable?

(a) Input data, $x \in \mathbb{R}$, computation of $2x$ as $x \oplus x$.

(b) Input data, $x \in \mathbb{R}$, computation of x^2 as $x \otimes x$.

(c) Input data, $x \in \mathbb{R} \setminus \{0\}$, computation of 1 as $x \oslash x$.

(d) Input data, $\mathbf{x}, \mathbf{y} \in \mathbb{R}^m$, computation of the inner product $\mathbf{x}^T \mathbf{y}$ as $(x_1 \otimes y_1) \oplus (x_2 \otimes y_2) \oplus (x_3 \otimes y_3) \oplus ...(x_m \otimes y_m)$.

(e) Input data $\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$, computation of eigen-values of $\mathbf{A}$ by evaluating the roots of characteristic polynomial.

[Hint: You need to examine the stability by looking at how the eigen-values of perturbed matrix $\mathbf{A} + \delta \mathbf{A}$ can be computed by finding the roots of the corresponding characteristic polynomial]