MA398 30th October, 2023.

Exercise Sheet 4

1. Prove that the condition number of a scalar function $f : \mathbb{R} \to \mathbb{R}$ is invariant under scaling of the function. That is, show that if $f(x) = ag(x)$ for some constant a and function g, then $\kappa_f(x) = \kappa_g(x)$.

   Hint: Use the fact that the derivative of $f(x)$ with respect to x is $f'(x) = ag'(x)$ and apply the definition of the condition number.

   Answer: Proof:

   Given the function $f(x) = ag(x)$, where a is a constant, we want to show that the condition number $\kappa_f(x) = \kappa_g(x)$.

   Recall the definition of the condition number:

$$\kappa_f(x) = \left\| \frac{xf'(x)}{f(x)} \right\|$$

   From lecture notes

   The derivative of $f(x)$ is $f'(x) = ag'(x)$. Therefore, we can express the condition number of f as:

$$\kappa_f(x) = \left\| \frac{xag'(x)}{ag(x)} \right\|$$

   The constant a can be cancelled from the numerator and the denominator:

$$\kappa_f(x) = \left\| \frac{xg'(x)}{g(x)} \right\|$$

   But this is just the condition number of g:

$$\boxed{\kappa_f(x) = \kappa_g(x)}$$

   This completes the proof. It shows that the condition number of a function is invariant under scaling – i.e., multiplying the function by a constant does not change its condition number.

3. Analyze the backward stability of the division operation in floating-point arithmetic similar to the subtraction operation shown in Example 1 in lecture 11. Assume that the exact input is $x = (x_1, x_2)^T$, and exact operation is $f(x) = x_1/x_2 = y$. Define and calculate the backward error, discuss the stability of this operation and any potential limitations or special cases that need to be considered.

   Answer: The exact operation is $x = (x_1, x_2)^T$ and $f(x) = x_1/x_2 = y$. Suppose we have the computed version as $\theta = fl(x_1) \oslash fl(x_2)$, where $\oslash$ represents the division operation in floating-point arithmetic and $fl(\cdot)$ represents the floating-point approximation of a number.

   In floating-point arithmetic, each operation is subject to a small rounding error. Let's denote these rounding errors as $\epsilon^{(1)}$ and $\epsilon^{(2)}$ for $fl(x_1)$ and $fl(x_2)$ respectively, with $|\epsilon^{(i)}| \le \epsilon_m$, $i = 1, 2$. Then we can write $fl(x_1) = x_1(1 + \epsilon^{(1)})$ and $fl(x_2) = x_2(1 + \epsilon^{(2)})$.

   Now, the computed division is $\theta = fl(x_1) \oslash fl(x_2) = \xi_1 \oslash \xi_2 = (\xi_1/\xi_2)(1 + \epsilon^{(3)})$, where $\epsilon^{(3)}$ is another rounding error introduced by the division operation.

   This simplifies to:

$$\frac{1+\varepsilon_1}{1+\varepsilon_2} \times \frac{1-\varepsilon_2}{1-\varepsilon_2} = \frac{1+\varepsilon_1-\varepsilon_2-\varepsilon_1\varepsilon_2}{1^2-\underbrace{\varepsilon_2^2}_{\to 0}}$$

$$\begin{aligned}
\theta &= (x_1(1 + \epsilon^{(1)})/x_2(1 + \epsilon^{(2)}))(1 + \epsilon^{(3)}) \\
&= (x_1/x_2)(1 + \epsilon^{(1)} - \epsilon^{(2)} + \epsilon^{(1)}\epsilon^{(2)})(1 + \epsilon^{(3)}) \\
&= y(1 + \epsilon^{(4)})
\end{aligned}$$

   where $\epsilon^{(4)} = \epsilon^{(1)} - \epsilon^{(2)} + \epsilon^{(1)}\epsilon^{(2)} + \epsilon^{(3)} + \epsilon^{(1)}\epsilon^{(3)} - \epsilon^{(2)}\epsilon^{(3)} + \epsilon^{(1)}\epsilon^{(2)}\epsilon^{(3)} = O(\epsilon_m)$ as $\epsilon_m \searrow 0$.

   Thus, the computed division can be written as an exact division with perturbed input data: $\theta = f(\zeta)$ for $\zeta = (x_1(1 + \epsilon^{(4)}), x_2)^T$. The absolute backward error is $|\zeta - x| = |(\epsilon^{(4)}x_1, 0)^T| \le \epsilon_m|x_1|$. The relative backward error is $|\zeta - x|/|x| \le \epsilon_m$.

   This means that the division operation is backward stable under the assumption that $x_2 \neq 0$.

5. (Growth factor) Show that

$$A = \begin{pmatrix} 1 & 0 & \cdots & 0 & 1 \\ -1 & \ddots & \ddots & \vdots & \vdots \\ \vdots & \ddots & \ddots & 0 & \vdots \\ \vdots & & \ddots & \ddots & \vdots \\ -1 & \cdots & \cdots & -1 & 1 \end{pmatrix}$$

has growth factor $g_n(A) = 2^{n-1}$. Hint: Denoting the U matrix in algorithm LU after step K by $U^{(k)}$, show by induction that $U^{(k)}_{n,n} = 2^k$.

Note that $g(n) \leq 2^{n-1}$ $\forall A \in \mathbb{C}^{n\times n}$ [Lemma 12.1 notes]

Numerical stability of the GE depends on the growth factor.

This is useful when we need to obtain a bound for the backward error of a computed solution, particularly as it concerns the unit round-off of the floating-point arithmetic. [Theorem 12.4 notes]

Note that a 4×4 example of the matrix A above is

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ -1 & 1 & 0 & 1 \\ -1 & -1 & 1 & 1 \\ -1 & -1 & -1 & 1 \end{bmatrix}.$$

Proceeding to reduce the above matrix to upper triangular:

$R_1 + R_2 \to R_2$
$R_1 + R_3 \to R_3$
$R_1 + R_4 \to R_4$

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & -1 & 1 & 2 \\ 0 & -1 & -1 & 2 \end{bmatrix}$$

$R_2 + R_3 \to R_3$
$R_2 + R_4 \to R_4$

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 4 \\ 0 & 0 & -1 & 4 \end{bmatrix}$$

$R_3 + R_4 \to R_4$

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 4 \\ 0 & 0 & 0 & 8 \end{bmatrix} =: U$$

In general, for some matrix

$$A = \begin{pmatrix} 1 & 0 & \cdots & 0 & 1 \\ -1 & \ddots & \ddots & \vdots & \vdots \\ \vdots & \ddots & \ddots & 0 & \vdots \\ \vdots & & \ddots & \ddots & \vdots \\ -1 & \cdots & \cdots & -1 & 1 \end{pmatrix}$$

$$U = \begin{bmatrix} 1 & 0 & 0 & \cdots & \cdots & 1 \\ 0 & 1 & 0 & \cdots & \cdots & 2 \\ 0 & 0 & 1 & \cdots & \cdots & 4 \\ 0 & 0 & 0 & 1 & \cdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & 2^{n-2} \\ 0 & 0 & 0 & \cdots & \cdots & 2^{n-1} \end{bmatrix}$$

$$g_n(A) = \frac{\|U\|_{max}}{\|A\|_{max}} = \frac{2^{n-1}}{1} = \underline{\underline{2^{n-1}}}$$

Definition 12.1