David Yang

due 11/8

Math 66 - Fall 2023 - Goldwyn

HW 6 - due Wednesday 11/8

(1) (a) For the matrix

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

use LU factorization with partial pivoting to find a permutation matrix P, a lower triangular matrix L, and an upper triangular matrix U so that $PA = LU$

(b) Use the matrices you found in (a) to solve

$$A\mathbf{x} = \begin{bmatrix} 0 \\ 3 \\ 5 \end{bmatrix}$$

Please: Show (clearly) your construction of P, L, U in (a) using elementary row operations, and your use of forward and backward substitution in (b).

Recommendation: Questions #1 - 4 in Sauer Chapter 2.4 are similar. I recommend you practice these. You can expect a question like this on the final exam.

(2) Use the matrix

$$A = \begin{bmatrix} \delta & 0 \\ 1 & 1 \end{bmatrix}$$

where δ is some very small number.

(a) find a formula for the exact solution to

$$A \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

your answer will be expressions for x_1 and x_2 that include the constants δ, b_1, and b_2.

(b) Using your answer to (a), describe the effect of any small changes to b_1 on the solution

(c) Based on your answer to (b), can you expect a numerical method working with floating point numbers to accurately solve this problem?

(d) Use the command `linalg.cond` in numpy to calculate the condition number of this matrix for $\delta = 1, 10^{-4}, 10^{-8}$, and 10^{-16}.

- Write out your results or include a screenshot.
- Is A ill-conditioned or not? How does this relate to your answer to (c)

(3) Let $\delta > 0$ be a small number and consider the interpolation points

$$(1, y_1) \quad \text{and} \quad (1+\delta, y_2) \quad \text{and} \quad (1+2\delta, y_3)$$

(a) Construct a 2nd degree interpolating polynomial in Newton's form (recall the last HW assignment where this was discussed). The coefficients of the interpolating polynomial will be expressions that include the y-values of the interpolation points.

(b) From your answer to (a), deduce that small changes in the y-values of the interpolation points can lead to large changes in the coefficients of the interpolating polynomial.

(c) If you were to use a **backward stable method** to solve this polynomial interpolation problem on a computer (with unavoidable rounding errors due to the use of floating point numbers), which one of the following would be true:

☐ the coefficients of the polynomial would be computed accurately

☑ evaluating the polynomial at x_1, x_2, and x_3 would produce values close to the values y_1, y_2, and y_3

☐ both of the above are true

☐ none of the above are true

Try to explain your choice using the phrases **forward error** and **backward error**.

1. (a)

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

Row operations on A:

$(R_1 \leftrightarrow R_2)$ implemented by $P_1 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ gives $P_1 A = \begin{bmatrix} 4 & -1 & 4 \\ 1 & -1 & 1 \\ 2 & 1 & 0 \end{bmatrix}$

$(-\frac{1}{4}R_1 + R_2 \to R_2)$ implemented by $L_1 = \begin{bmatrix} 1 & 0 & 0 \\ -\frac{1}{4} & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ gives $L_1 P_1 A = \begin{bmatrix} 4 & -1 & 4 \\ 0 & -\frac{3}{4} & 0 \\ 2 & 1 & 0 \end{bmatrix}$

$(-\frac{1}{2}R_1 + R_3 \to R_3)$ implemented by $L_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -\frac{1}{2} & 0 & 1 \end{bmatrix}$ gives $L_2 L_1 P_1 A = \begin{bmatrix} 4 & -1 & 4 \\ 0 & -\frac{3}{4} & 0 \\ 0 & \frac{3}{2} & -2 \end{bmatrix}$

$(2R_2 + R_3 \to R_3)$ implemented by $L_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 2 & 1 \end{bmatrix}$ gives $L_3 L_2 L_1 P_1 A = \begin{bmatrix} 4 & -1 & 4 \\ 0 & -\frac{3}{4} & 0 \\ 0 & 0 & -2 \end{bmatrix} \equiv U.$

Note that $P_1 A = \underbrace{L_1^{-1} L_2^{-1} L_3^{-1}}_{L} U$, where

$$L = L_1^{-1} L_2^{-1} L_3^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{4} & 1 & 0 \\ \frac{1}{2} & -2 & 1 \end{bmatrix} \quad \text{and} \quad P = P_1 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

We can check that $PA = LU$ as intended.

(b) From (a), we have $PA = LU$. To solve $Ax = b$, where $b = \begin{bmatrix} 0 \\ 3 \\ 5 \end{bmatrix}$, we will first solve $L\vec{y} = P\vec{b}$ and then solve $U\vec{x} = \vec{y}$.

$$L\vec{y} = P\vec{b} \Rightarrow \begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{4} & 1 & 0 \\ \frac{1}{2} & -2 & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 3 \\ 0 \\ 5 \end{bmatrix}$$ as $P: R_1 \leftrightarrow R_2$ of b

Using forward substitution, we find $y_1 = 3$, $y_2 = 0 - \frac{1}{4}y_1 = \frac{-3}{4}$, $y_3 = 5 + 2y_2 - \frac{1}{2}y_1 = 2$, so $\vec{y} = \begin{bmatrix} 3 \\ -3/4 \\ 2 \end{bmatrix}$.

It remains to solve $U\vec{x} = \vec{y}$, i.e. $\begin{bmatrix} 4 & -1 & 4 \\ 0 & -\frac{3}{4} & 0 \\ 0 & 0 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 3 \\ -\frac{3}{4} \\ 2 \end{bmatrix}$.

By backwards substitution, we have $x_3 = \frac{2}{-2} = -1$, $x_2 = -\frac{3}{4} / -\frac{3}{4} = 1$, $x_1 = \frac{3 + x_2 - 4x_3}{4} = 2$.

Thus, $\boxed{\vec{x} = \begin{bmatrix} 2 \\ 1 \\ -1 \end{bmatrix}}$ solves the equation $A\vec{x} = \vec{b}$, which we can confirm by plugging this in.

2. $A = \begin{bmatrix} \delta & 0 \\ 1 & 1 \end{bmatrix}$

(a) $A \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$

$$\begin{bmatrix} \delta & 0 \\ 1 & 1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \Rightarrow \begin{array}{l} \delta x_1 = b_1 \\ x_1 + x_2 = b_2 \end{array}$$

Solving the system for x_1 and x_2, we get that $x_1 = \frac{b_1}{\delta}$, and $x_2 = b_2 - x_1 = b_2 - \frac{b_1}{\delta}$

So the exact solution is

$$\boxed{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \frac{b_1}{\delta} \\ b_2 - \frac{b_1}{\delta} \end{bmatrix}}$$

(b) For a very small δ, a small change in b_1 could drastically alter the solution:
$x_1 = \frac{b_1}{\delta}$ changes a lot if $\delta <<$ change in b_1 and likewise for
$x_2 = b_2 - \frac{b_1}{\delta}$ by the above logic.

(c) A numerical method working with floating point numbers may not accurately solve the problem.
Floating point numbers and operations on them may lead to rounding errors, and as we know from part (b), there small errors can lead to drastically different solutions to the system.

(d)

Condition number of $A = \begin{bmatrix} \delta & 0 \\ 1 & 1 \end{bmatrix}$	value of δ
≈ 2.618	1
20,000	10^{-4}
2×10^8	10^{-8}
overflow (inf)	10^{-16}

We notice that for small values of δ, the condition number is very large.
Equivalently, we know the relative forward error can be big even if backward error is small, and this matches with our answer from part (c), as a numerical method working with floating point numbers will have rounding errors that drastically alter the solution to the problem.

3. $\delta > 0$ small with interpolation points $(1, y_1)$, $(1+\delta, y_2)$, $(1+2\delta, y_3)$.

(a) 2^{nd} degree interpolating polynomial in Newton's form (passing through $(1,y_1)$ and $(1+\delta, y_2)$) is

$$y = a_0 + a_1(x-1) + a_2(x-1)(x-(1+\delta)).$$

Plugging in the points $(1,y_1)$, $(1+\delta, y_2)$, and $(1+2\delta, y_3)$, we have

$$\begin{cases} y_1 = a_0 + a_1(1-1) + a_2(1-1)(1-(1+\delta)) \\ y_2 = a_0 + a_1(1+\delta-1) + a_2(1+\delta-1)(1+\delta-(1+\delta)) \\ y_3 = a_0 + a_1(1+2\delta-1) + a_2(1+2\delta-1)(1+2\delta-(1+\delta)) \end{cases} \Rightarrow \begin{array}{l} y_1 = a_0 \\ y_2 = a_0 + a_1\delta \\ y_3 = a_0 + a_1(2\delta) + a_2(2\delta)(\delta). \end{array}$$

Solving, we get $a_0 = y_1$, $a_1 = \frac{y_2 - a_0}{\delta} = \frac{y_2 - y_1}{\delta}$, and $a_2 = \frac{y_3 - a_0 - a_1(2\delta)}{2\delta^2} = \frac{y_3 - y_1 - 2(y_2 - y_1)}{2\delta^2} = \frac{y_3 + y_1 - 2y_2}{2\delta^2}$

Thus, the 2^{nd} degree interpolating polynomial in Newton's form is

$$\boxed{y = y_1 + \frac{y_2 - y_1}{\delta}(x-1) + \frac{y_3 + y_1 - 2y_2}{2\delta^2}(x-1)(x-(1+\delta))}.$$

(b) Looking at the above interpolating polynomial, we notice that a small change in the y-values y_1, y_2, y_3 of the interpolation points will be "amplified" in the coefficients of the interpolating polynomial. Since δ is very small, $\frac{y_2 - y_1}{\delta}$ and $\frac{y_3 + y_1 - 2y_2}{2\delta^2}$ may change drastically as a small change in y_2 or y_3 could be $\gg \delta$ or δ^2, and the divisions by $\delta, 2\delta^2$ may lead to large changes in the coefficients.

(c) A backwards stable method is one that guarantees small backward error.
In this case, for a computed solution $\hat{a} = \begin{bmatrix} \hat{a}_0 \\ \hat{a}_1 \\ \hat{a}_2 \end{bmatrix}$ to the equation $\begin{bmatrix} 1 & 0 & 0 \\ 1 & \delta & 0 \\ 1 & 2\delta & 2\delta^2 \end{bmatrix} a = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$,
the backwards error is

$$\left\| \underbrace{\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}}_{\text{true}} - \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 1 & \delta & 0 \\ 1 & 2\delta & 2\delta^2 \end{bmatrix} \hat{a}}_{\text{Computed } \hat{y}_1, \hat{y}_2, \hat{y}_3} \right\|.$$

If this backward error is small, we know that $\boxed{\text{evaluating polynomials at } x_1, x_2, x_3 \text{ would produce values close to } y_1, y_2, y_3}$

On the other hand, we know from part (b) even small changes in y-values may lead to large changes in the coefficients of the interpolating polynomial. Thus, we cannot make any such conclusion about the forward error being minimized, so we cannot guarantee the coefficients would be computed accurately.