3. LU-factorization of a matrix. Presentation of the algorithm. Discussion of pivoting and pivoting strategies. Give examples using `lugui`.

Numerical Analysis E2021

Institute of Mathematics
Aalborg University

AALBORG UNIVERSITY
DENMARK

Motivation

- Inverting a matrix is literally a breach on several human rights acts
  - problems with condition number
  - efficiency and accuracy
  - memory usage
- One method, among many, to dodge this is using LU factorisation

LU Factorisation

Any square-matrix A can be written as PLU using elementary row operations.

Elementary row operations can easily be represented using elementary matrices, and their inverses are also easily determined. Thus,

$$U = E_n E_{n-1} \dots E_1 C \tag{1}$$

$$L = E_1^{-1} \dots E_{n-1}^{-1} E_n^{-1} \tag{2}$$

Using this we can easily solve a given linear system

$$Cx = LUx = b \tag{3}$$

by using forward substitution on $Lc = b$, and backwards substitution on $Ux = c$.

Numerical Analysis
E2021

Motivation
LU Factorisation
Pivoting 3
General LU-algorithm
Pivoting strategies

Consider

$$A = \begin{bmatrix} 10^{-20} & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 10^{20} & 1 \end{bmatrix} \begin{bmatrix} 10^{-20} & 1 \\ 0 & 1 - 10^{20} \end{bmatrix} = LU \tag{4}$$

Assuming we do this in floating-point arithmetic, then the number $1 - 10^{20}$ can be represented as -10^{20} on a computer with $\varepsilon_m = 10^{-16}$. The floating-point representations will then be

$$\tilde{L} = \begin{bmatrix} 1 & 0 \\ 10^{20} & 1 \end{bmatrix}, \quad \tilde{U} = \begin{bmatrix} 10^{-20} & 1 \\ 0 & -10^{20} \end{bmatrix} \tag{5}$$

but

$$\tilde{L}\tilde{U} = \begin{bmatrix} 10^{-20} & 1 \\ 1 & 0 \end{bmatrix} \tag{6}$$

Remedy to aforementioned problem will be pivoting, where we, in general, will attempt to make our pivots as large as possible.

General LU-algorithm

The following method describes LU-factorisation without permutation / row exchanges.

```
for j = 1 to min{m, n} do
    for i = j to m do
        if a_ii ≠ 0 then
            for k = i + 1 to m do
                Multiply row i by −a_ki/a_ii and add to row k
        end if
        if a_ii = 0 then
            for k = i + 1 to m do
                if a_ki ≠ 0 then
                    STOP
        end if
```

If one encounters the statement **STOP** then A does not have an LU factorisation without row exchanges.

To carry out the LU factorisation with row exchanges one then exchanges rows i and k each time the **STOP** statement is encountered, and then returns to the previous **if** statement for the column below the new row i.

Pivoting strategies

Several pivoting strategies can be employed.

- Diagonal pivoting
  - The diagonal element is used as the pivot
- Complete pivoting
  - Largest element in absolute value in unaltered submatrix is used as pivot
- Partial pivoting
  - Largest element in absolute value in unreduced part of the current column
- Manual pivoting

MATLAB demo of `lugui`