Chapter 5

GAUSSIAN ELIMINATION AND LU FACTORIZATIONS

ch5

Background Material Needed

- Vector and matrix norms and their properties (**Section 2.5**)
- Special matrices (**Section 2.4**)
- Concepts of errors, floating point operations, and stability (**Section 3.2**, **Sections 4.2 and 4.3**)

5.1 A Computational Template in Numerical Linear Algebra

Most computational algorithms to be presented in this book have a common basic structure that can be described in the following three steps:

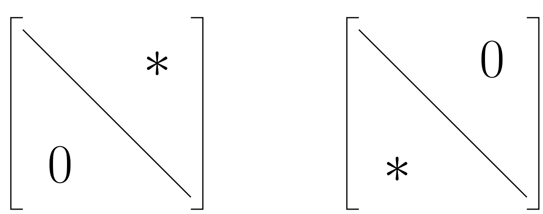
Step 1. The problem is first transformed to an "easier-to-solve" problem, by transforming the associated matrices to "condensed" forms with special structures.

Step 2. The transformed problem is then solved by exploiting the special structures of these condensed forms.

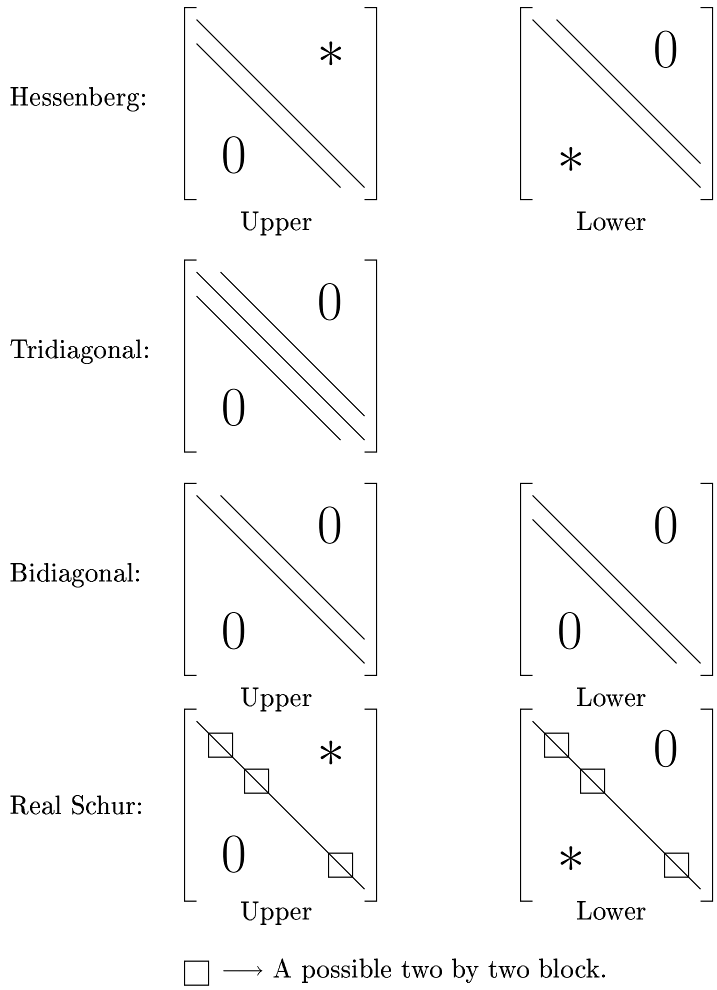
Step 3. Finally, the solution of the original problem is recovered from the solution of the transformed problem.

The typical **condensed** forms are:

Triangular:



Upper Lower



- The system of linear equation $Ax = b$ is solved by transforming A to an upper triangular matrix (Gaussian elimination), followed by solving two triangular systems: upper and lower (**Chapter 6**).
- The eigenvalues of a matrix A are computed by transforming A first to an upper Hessenberg matrix H, followed by reducing H further to a Real Schur matrix iteratively (**Chapter 9**).
- The singular values of A are computed by transforming A first to a bidiagonal matrix followed by further reduction of the bidiagonal matrix to a diagonal matrix (**Chapters 7 and 10**).

5.2 LU Factorization Using Gaussian Elimination

The tools of Gaussian elimination are **elementary matrices**.

D5-1 **Definition 5.1.** *An elementary lower triangular matrix of order n of type k is a matrix of the form*

$$M_k = \begin{pmatrix} 1 & 0 & \cdots & \cdots & \cdots & 0 & 0 & 0 \\ 0 & 1 & 0 & \cdots & \cdots & \cdots & 0 & 0 \\ 0 & 0 & 1 & 0 & & & & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & & & \vdots \\ \vdots & \vdots & & 0 & 1 & \ddots & & \vdots \\ 0 & \vdots & & & m_{k+1,k} & \ddots & 0 & \vdots \\ 0 & 0 & & & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & \cdots & m_{n,k} & \cdots & 0 & 1 \end{pmatrix} \tag{5.1}$$ eq5.2.1

Thus, *it is an identity matrix except possibly for a few nonzero elements below the diagonal of the k^{th} column.* The matrix M_k can be written in the form **[Exercise 5.1(a)]**:

$$M_k = I + m_k e_k^T,$$

where I is the identity matrix of order n, $m_k = (0, 0, \ldots, 0, m_{k+1,k}, \ldots, m_{n,k})^T$, and e_k is the k^{th} unit-vector, that is, $e_k^T = (0, 0, \ldots,\ 0,\ 1, 0, \cdots, 0)$, where '1' is at the kth entry.

5.2.1 Creating Zeros in a Vector or Matrix using Elementary Matrix

Lemma 5.2. *Let*

$$a_1 = \begin{pmatrix} a_{11} \\ a_{21} \\ \vdots \\ a_{n1} \end{pmatrix}.$$

Then the elementary matrix

$$M_1 = \begin{pmatrix} 1 & 0 & 0 & \cdots & \cdots & 0 \\ \frac{-a_{21}}{a_{11}} & 1 & 0 & \cdots & \cdots & 0 \\ \frac{-a_{31}}{a_{11}} & 0 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & & \vdots \\ \frac{-a_{n1}}{a_{11}} & 0 \cdots & 0 & 0 & & 1 \end{pmatrix} \tag{5.2}$$ 5.2.2

is such that

$$M_1 a_1 = \begin{pmatrix} a_{11} \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \qquad (5.3)$$ 5.2.3

Proof. **[Exercise 5.1(c)].** □

E5.2.1 **Example 5.3** Let

$$a = \begin{pmatrix} 2 \\ 5 \\ 1 \\ 2 \end{pmatrix}.$$

Then

$$M_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -\frac{5}{2} & 1 & 0 & 0 \\ -\frac{1}{2} & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{pmatrix}$$

is such that

$$M_1 a = \begin{pmatrix} 2 \\ 0 \\ 0 \\ 0 \end{pmatrix}. \quad \blacksquare$$

5.2.2 Triangularization Using Gaussian Elimination

The elementary matrices can be conveniently used in triangularizing a matrix. The process is called *Gaussian Elimination, after the name of the famous German mathematician and astronomer, Karl Friedrich Gauss*[*].

Given an $n \times n$ matrix A, Gaussian elimination process consists of finding the elementary matrices $M_1, \cdots, M_{n-1}$ such that

- $A^{(1)} = M_1 A$ has zeros in the first column below the (1,1) entry.
- $A^{(2)} = M_2 A^{(1)}$ has zeros in the second column below (2,2) entry.
- $A^{(n-1)} = M_{n-1} A^{(n-2)}$ has zeros in the $(n-1)^{th}$ column below the $(n-1, n-1)$ entry.

[*]**Karl Friedrich Gauss** (1777-1855) was a German mathematician and astronomer, noted for development of many classical mathematical theories, and for his calculation of the orbits of the asteroids Ceres and Pallas. Gauss is still regarded as one of the greatest mathematicians the world has ever produced.

The final matrix $A^{(n-1)}$ is upper triangular. The key observation is that each of the matrices $A^{(k)}$ is the result of the premultiplication of $A^{(k-1)}$ by an elementary matrix.

The following is an illustrative diagram in case $n = 4$.

Step 1. $$A \xrightarrow{M_1} M_1 A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & a_{24}^{(1)} \\ 0 & a_{32}^{(1)} & a_{33}^{(1)} & a_{34}^{(1)} \\ 0 & a_{42}^{(1)} & a_{43}^{(1)} & a_{44}^{(1)} \end{pmatrix} = A^{(1)}$$

Step 2. $$A^{(1)} \xrightarrow{M_2} M_2 A^{(1)} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & a_{24}^{(1)} \\ 0 & 0 & a_{33}^{(2)} & a_{34}^{(2)} \\ 0 & 0 & a_{43}^{(2)} & a_{44}^{(2)} \end{pmatrix} = A^{(2)}$$

Step 3. $$A^{(2)} \xrightarrow{M_3} M_3 A^{(2)} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & a_{24}^{(1)} \\ 0 & 0 & a_{33}^{(2)} & a_{34}^{(2)} \\ 0 & 0 & 0 & a_{44}^{(3)} \end{pmatrix} = A^{(3)}$$

Figure 5.1 Illustration of Gaussian Elimination

Notes:

(i) The matrix $A^{(1)}$ in Step 1 can be formed as:

- Find M_1 such that $M_1 \begin{pmatrix} a_{11} \\ a_{12} \\ a_{13} \\ a_{14} \end{pmatrix} = \begin{pmatrix} a_{11} \\ 0 \\ 0 \\ 0 \end{pmatrix}$
- Update: $A^{(1)} = M_1 A$

(ii) The matrix $A^{(2)}$ in Step 2 can be formed in two smaller steps as follows:

- Find $\hat{M}_2$ such that $\hat{M}_2 \begin{pmatrix} a_{22}^{(1)} \\ a_{32}^{(1)} \\ a_{42}^{(1)} \end{pmatrix} = \begin{pmatrix} a_{22}^{(2)} \\ 0 \\ 0 \end{pmatrix}$

- Form $M_2 = \begin{pmatrix} 1 & 0 \\ 0 & \hat{M}_2 \end{pmatrix}$
- Update $A^{(2)} = M_2 A^{(1)}$

and so on.

(iii) In practice, *neither the matrices* M_k *nor the products* $M_k A^{(k-1)}$ *need to be explicitly formed*, as shown below with a 4×4 *numerical example.*

E5.2.2 **Example 5.4** Consider

$$A = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 1 & 1 & 3 & 3 \\ 2 & 1 & 1 & 1 \end{pmatrix}.$$

Step 1. (Eliminate the entries of the 1st column of A below the diagonal). Multiply the 1st row by $-5, -1, -2$, and add respectively, to the 2nd through 4th rows. At the end of this step, we have

$$A^{(1)} = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 0 & -4 & -8 & -12 \\ 0 & -1 & 0 & -1 \\ 0 & -3 & -5 & -7 \end{pmatrix}.$$

Note that in terms of the matrix multiplication, we have

$$A^{(1)} = \underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 \\ -5 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ -2 & 0 & 0 & 1 \end{pmatrix}}_{M_1} \underbrace{\begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 1 & 1 & 3 & 3 \\ 2 & 1 & 1 & 1 \end{pmatrix}}_{A} = M_1 A.$$

Step 2. (Eliminate the entries of the 2nd column of $A^{(1)}$ below the diagonal). Multiply the 2nd row of $A^{(1)}$ by $-\frac{1}{4}$ and $-\frac{3}{4}$ and add respectively, to the 3rd and 4th rows. At the end of this step, we have

$$A^{(2)} = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 0 & -4 & -8 & -12 \\ 0 & 0 & 2 & 2 \\ 0 & 0 & 1 & 2 \end{pmatrix}.$$

Again, in terms of matrix multiplication, we have

$$A^{(2)} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -\frac{1}{4} & 1 & 0 \\ 0 & -\frac{3}{4} & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 2 & 3 & 4 \\ 0 & -4 & -8 & -12 \\ 0 & -1 & 0 & -1 \\ 0 & -3 & -5 & -7 \end{pmatrix} = M_2 A^{(1)}.$$

Step 3. (Eliminate the entries of the third column of $A^{(2)}$ below the diagonal). Multiply the third column of $A^{(2)}$ by $-\dfrac{1}{2}$ and add it to the third, giving:

$$A^{(3)} = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 0 & -4 & -8 & -12 \\ 0 & 0 & 2 & 2 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \text{Upper triangular.}$$

Again,

$$A^{(3)} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{2} & 1 \end{pmatrix} \begin{pmatrix} 1 & 2 & 3 & 4 \\ 0 & -4 & -8 & -12 \\ 0 & 0 & 2 & 2 \\ 0 & 0 & 1 & 2 \end{pmatrix} = M_3 A^{(2)}.$$

■

Remark: *Note that to form $A^{(k)}$ from $A^{(k-1)}, k = 1, 2, 3$, neither the matrices M_k nor the products $M_k A^{(k-1)}$ need to be performed explicitly.*

The general process is now quite clear.

Starting with A, the process constructs successively the matrices $A^{(1)}$, $A^{(2)}$, $\ldots$, $A^{(n-1)}$ such that $A^{(1)}$ has zeros on the 1st column below the diagonal, $A^{(2)}$ has zeros on the second column below its diagonal and so on. The final matrix $A^{(n-1)}$ is an upper triangular matrix. *The key observation is that each of these matrices is a result of premultiplication of the previous one by an elementary lower triangular matrix.*

General process. There are $(n-1)$ steps. Let $A^{(k)} = (a_{ij}^{(k)}),\ k \geq 1$.

Step 1. (Eliminate the entries of the first column of A below the diagonal). Multiply the entries of the 1st row of A by the numbers

$$m_{i1} = -\frac{a_{i1}}{a_{11}},\ i = 2, \ldots, n$$

and add them, respectively, to those of the second through n^{th} row. We have a new matrix $A^{(1)}$:

$$A^{(1)} = \begin{pmatrix} a_{11} & a_{12} & \cdots & \cdots & a_{1n} \\ 0 & a_{22}^{(1)} & \cdots & \cdots & a_{2n}^{(1)} \\ 0 & \vdots & \ddots & & \vdots \\ \vdots & & & \ddots & \vdots \\ 0 & a_{n2}^{(1)} & \cdots & \cdots & a_{nn}^{(1)} \end{pmatrix},$$

which can be written as:

$$A^{(1)} = \begin{pmatrix} 1 & 0 & 0 & \ldots & 0 \\ m_{21} & 1 & 0 & \ldots & 0 \\ \vdots & & \ddots & & \\ m_{n1} & 0 & \ldots & 0 & 1 \end{pmatrix} \begin{pmatrix} a_{11} & \ldots & a_{1n} \\ a_{21} & \ldots & a_{22} \\ \vdots & & \\ a_{n1} & \ldots & a_{nn} \end{pmatrix} = M_1 A$$

Step 2. (Eliminate the entries of the second column of $A^{(1)}$ below the diagonal). Multiply the entries of 2nd row of $A^{(1)}$ by the numbers

$$m_{i2} = -\frac{a_{i2}^{(1)}}{a_{22}^{(1)}}, i = 3, \ldots, n$$

and add them, respectively, to those of 3rd through n^{th} row.

We now have a new matrix $A^{(2)}$:

$$A^{(2)} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \ldots & a_{1n} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & \ldots & a_{2n}^{(1)} \\ 0 & 0 & a_{32}^{(2)} & \ldots & a_{3n}^{(2)} \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & a_{n3}^{(2)} & \ldots & a_{nn}^{(2)} \end{pmatrix},$$

which can be written as:

$$A^{(2)} = \begin{pmatrix} 1 & & & & \\ 0 & 1 & & & 0 \\ 0 & m_{32} & \ddots & & \\ \vdots & \vdots & & \ddots & \\ 0 & m_{n2} & \ldots & \ldots & 1 \end{pmatrix} \begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ 0 & a_{22}^{(1)} & \ldots & a_{2n}^{(1)} \\ 0 & a_{32}^{(1)} & \ldots & a_{3n}^{(1)} \\ \vdots & \vdots & & \\ 0 & a_{n2}^{(1)} & \ldots & a_{nn}^{(1)} \end{pmatrix} = M_2 A^{(1)}.$$

The process is fairly general. The general k^{th} step can now easily be written down.

Step k $(k > 1)$. **(eliminate the entries below the diagonal of the k^{th} column of $A^{(k-1)}$.)** Multiply the k^{th} row of $A^{(k-1)}$ by the numbers

$$m_{ik} = -\frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}}, \quad i = k+1, \ldots, n$$

and add, respectively, to the $(k+1)^{th}$ through n^{th} row. This will yield a matrix $A^{(k)}$ given by

$$A^{(k)} = M_k A^{(k-1)}$$

where M_k is defined by (5.1). eq5.2.1

Step n-1. At the end of the $(n-1)^{th}$ step, the matrix $A^{(n-1)}$ is upper triangular:

$$A^{(n-1)} = \begin{pmatrix} a_{11} & a_{12} & \cdots & \cdots & \cdots & a_{1n} \\ 0 & a_{22}^{(1)} & \cdots & \cdots & \cdots & a_{2n}^{(1)} \\ 0 & 0 & a_{33}^{(2)} & & & a_{3n}^{(2)} \\ 0 & & & \ddots & & \vdots \\ \vdots & & & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & a_{nn}^{(n-1)} \end{pmatrix}.$$

Similar to the other steps, we have (**in terms of matrix multiplications**):

$$A^{(n-1)} = M_{n-1}A^{(n-2)}.$$

The LU Factorization of a Matrix from Gaussian Elimination

The process we just described yields a factorization of the matrix: $A = LU$, where L is *unit lower triangular* and U is *upper triangular*, as shown below. This factorization is known as *LU factorization of A*. *LU factorization of a matrix is an important matrix factorization useful for solving a linear system and computing the determinant and the inverse of a matrix (see **Chapter 6**).*

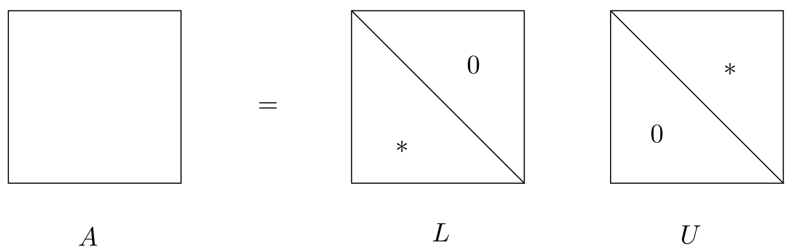


Figure 5.2 LU Factorization of a Matrix

Obtaining L and U

First, observe that the final matrix $A^{(n-1)}$ is an upper triangular matrix. So, we can take this matrix as our U matrix.

Thus $U = A^{(n-1)} = M_{n-1}A^{(n-2)}$. Again, $A^{n-2} = M_{n-2}A^{(n-3)}$.

So, $U = M_{n-1}M_{n-2}A^{(n-3)}$. Continuing this way, we can write

$$U = M_{n-1}M_{n-2}\dots M_2M_1A$$

Set now $L_1 = M_{n-1}M_{n-2}\dots M_2M_1$. Then $U = L_1A$.

Since each of the elementary lower triangular matrices is a unit lower triangular matrix(*a lower triangular matrix with 1's along the diagonal)* it follows that L_1 is invertible and L_1^{-1} is also **unit lower triangular.** (Note that *the product of unit lower triangular matrices is a unit lower triangular matrix and so is the inverse).* Now set $L = L_1^{-1}$. Then $A = LU$.

LU Factorization of a Matrix from Gaussian Elimination

$$A = LU$$

- $L = (M_{n-1}M_{n-2}\dots M_2M_1)^{-1}$ (Unit Lower Triangular).
- $U = A^{n-1}$ (Upper Triangular).

D5-3 **Definition 5.5.** *The entries $a_{11}, a_{22}^{(1)}, \dots, a_{nn}^{(n-1)}$ are called pivots, and the above process of obtaining LU factorization is known as Gaussian elimination without row* *interchanges. It is commonly known as Gaussian elimination without pivoting. The numbers m_{ik} are called multipliers.*

Obtaining L without Matrix Inversion

We will now show that the matrix L can be formed without explicitly computing any matrix product and without any matrix inversion.

$$L = L_1^{-1} = M_1^{-1} M_2^{-1} \dots M_{n-1}^{-1}.$$

First, we observe [**Exercise 5.1(b)**] that

$$M_i^{-1} = I - m e_i^T, \; i = 1, 2, \dots, n-1$$

where $m = (0, 0, \dots, 0, m_{i+1,i} \dots, m_{n,i})^T$ and e_i is the i^{th} unit vector.

It simply means that the M_i^{-1} is just the matrix M_i except that the entries on the i^{th} column below the diagonal are just the negatives of the corresponding entries of M_i.

Thus, $M_i^{-1} M_{i+1}^{-1}$ is a unit lower triangular matrix with the nonzero entries below the diagonal only on the columns i and $(i+1)$, which are the negatives of the corresponding entries of M_i and M_{i+1}. For example,

$$M_1^{-1} M_2^{-1} = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ -m_{21} & 1 & & \dots & 0 \\ -m_{31} & -m_{32} & \ddots & & \vdots \\ \vdots & \vdots & 0 & \ddots & \\ -m_{n1} & -m_{n2} & \dots & \dots & 1 \end{pmatrix}.$$

This implies that

$$L = M_1^{-1} M_2^{-1} \dots M_{n-1}^{-1} = \begin{pmatrix} 1 & 0 & \cdots & \cdots & 0 \\ -m_{21} & 1 & 0 & & 0 \\ -m_{31} & -m_{32} & 1 & \ddots & 0 \\ \vdots & \vdots & \ddots & \ddots & 0 \\ -m_{n1} & -m_{n2} & \cdots & -m_{n,n-1} & 1 \end{pmatrix}.$$

Thus to form L do the following:

- Save the multipliers at each step.
- Insert the negative of the multipliers of Step 1 in the first column of the identity matrix below the diagonal, the negatives of the multipliers of Step 2 in the second column below the diagonal, and so on.

E5.2.3 **Example 5.6**

$$A = \begin{pmatrix} 2 & 2 & 3 \\ 4 & 5 & 6 \\ 1 & 2 & 4 \end{pmatrix}.$$

Step 1. (Eliminate the entries on the first column of A below the diagonal). Multiply the first row of A by -2 and $-\frac{1}{2}$ and add, respectively, to the second and third row.

$$A^{(1)} = \begin{pmatrix} 2 & 2 & 3 \\ 0 & 1 & 0 \\ 0 & 1 & -\frac{5}{2} \end{pmatrix}; m_{21} = -2, \; m_{31} = -\tfrac{1}{2}.$$

Step 2. (Eliminate the entries on the second column of $A^{(1)}$ below the diagonal.) Multiply the second row of $A^{(1)}$ by -1 and add it to the third row.

$$A^{(2)} = \begin{pmatrix} 2 & 2 & 3 \\ 0 & 1 & 0 \\ 0 & 0 & \frac{5}{2} \end{pmatrix}; m_{32} = -1.$$

So,

$$U = A^{(2)} = \begin{pmatrix} 2 & 2 & 3 \\ 0 & 1 & 0 \\ 0 & 0 & \frac{5}{2} \end{pmatrix}.$$

$$L = L_1^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ -m_{21} & 1 & 0 \\ -m_{31} & -m_{32} & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ \frac{1}{2} & 1 & 1 \end{pmatrix}$$

(Note that neither L_1 nor its inverse needs to be computed explicitly.)
■

Existence and Uniqueness of LU Factorization

Note that for an LU factorization to exist, the pivots must be different from zero. Thus, LU factorization may not exist even for a very simple matrix. Take $A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$. The pivot $a_{11}^{(0)}$ is zero. So, *Gaussian elimination scheme can not be carried out.*

The following theorem gives conditions on the existence and uniqueness of *LU* factorization.

Definition 5.7. *The k-th leading principal minor of a matrix, A, denoted by A_k, is defined to be the $k \times k$ leading principal submatrix consisting of the first k rows and the first k columns.*

T5.2.1 **Theorem 5.8.** ***(Existence and Uniqueness of LU Factorization)***

(i) An $n \times n$ matrix A has an LU factorization if $A_k, k = 1, \ldots, n-1$, are nonsingular.

(ii) If the LU factorization exists and A is nonsingular, then this factorization is unique.

Proof. **Existence:** From the derivation of Gaussian elimination scheme, it follows that the process can only break down if any of the pivots $a_{11},\ a_{22}^{(1)}, \ldots, a_{nn}^{(n-1)}$ is zero. Again, it can be shown [**Exercise 5.3**] that

$$\det(A_k) = a_{11}a_{22}^{(1)} \ldots a_{k}^{(k-1)}, k = 1, \ldots, n-1 \text{ (Note that } \det A_1 = a_{11}).$$

This means that if the first $(n-1)$ leading principal minors are nonsingular, then Gaussian elimination scheme does not fail, and we always have an LU factorization of A in this case, as shown by the above discussion.

Uniqueness: The uniqueness will be proved by **contradiction**. Suppose there are two different LU factorizations of A: $A = L_1U_1 = L_2U_2$. Then, we must show that $L_1 = L_2$ and $U_1 = U_2$.
Because A is nonsingular, the matrices L_1, L_2, U_1 and U_2 are all nonsingular. Thus it follows from the above two factorizations of A, $L_2^{-1}L_1 = U_2U_1^{-1}$.

Now $L_2L_1^{-1}$ is a unit lower triangular matrix and $U_2U_1^{-1}$ is an upper triangular matrix and the only way they can be equal is that both of these are the identity. Thus $L_1 = L_2$ and $U_1 = U_2$. □

Remark: Note that in the above theorem, if the diagonal entries of L are not specified, then the factorization is <u>not</u> unique. (Do an example to verify this).

A Storage Scheme for a Practical LU Factorization

As Example E5.2.3 ~~5.6~~ shows that for a practical Gaussian elimination scheme, the k^{th}-step consists of

- Forming the multipliers $m_{ik} = -\dfrac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}}, i = k+1, \cdots, n$
- Updating the entries of the submatrix: $A(k+1:n, k+1:n)$ [the submatrix consisting of the rows $(k+1)$ through n and columns $(k+1)$ through n].

  The following storage scheme thus can be used:
- The multipliers are stored below the main diagonal of A. These multipliers then can be used to form L.
- The entries of the upper triangular matrix U are stored in the upper half part of A including the diagonal.

  With this storage scheme, the upper triangular matrix $A^{(n-1)}$ at the end of the $(n-1)^{th}$ step will look like:

$$A \equiv A^{(n-1)} = \begin{pmatrix} a_{11} & a_{12} & \cdots & \cdots & a_{1n} \\ m_{21} & a_{21}^{(1)} & \cdots & \cdots & a_{2n}^{(1)} \\ \vdots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \vdots \\ m_{n1} & \cdots & \cdots & m_{n,n-1} & a_{nn}^{(n-1)} \end{pmatrix}$$

ALGORITHM 5.1. LU Factorization using Gaussian Elimination without Pivoting.

A5.2.2

Input: An $n \times n$ matrix A
Outputs: (i) An upper triangular matrix U, and (ii) The multipliers m_{ij} needed to form the unit lower triangular matrix L such that $A = LU$.
Storage: The upper triangular part of U is stored over the upper triangular part of A including the diagonal. The multipliers needed to compute L are stored in the lower triangular part of L below the diagonal.

For $k = 1, 2, \ldots, (n-1)$ do

1. **(Form the multipliers)**:
   $a_{ik} \equiv m_{ik} = -\dfrac{a_{ik}}{a_{kk}}$ $(i = k+1, k+2, \ldots, n)$
2. (**Update the entries** of $A(k+1:n, k+1:n)$) :
   $a_{ij} \equiv a_{ij} + m_{ik}a_{kj} (i = k+1, \ldots, n; j = k+1, n)$.

End

Remark: The algorithm does not give the matrix L explicitly; however, it can be formed out of the multipliers saved at each step, as shown earlier (**see the expression for** L).

Example 5.9 Consider **Example** [E5.2.3] **5.6** again.

$\boldsymbol{k = 1}$ Multipliers are: $m_{21} = -2,\ m_{31} = -\frac{1}{2}$

Updated A: $A \equiv \begin{pmatrix} 2 & 2 & 3 \\ 0 & 1 & 0 \\ 0 & 1 & -\frac{5}{2} \end{pmatrix}$

$\boldsymbol{k = 2}$ Multiplier: $m_{32} = -1$

Updated $A \equiv$ $\begin{pmatrix} 2 & 2 & 3 \\ \boxed{0} & 1 & 0 \\ \boxed{0} & 0 & \frac{5}{2} \end{pmatrix}$

Form L and U: $L = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ \frac{1}{2} & 1 & 1 \end{pmatrix}$, $U = \begin{pmatrix} 2 & 2 & 3 \\ 0 & 1 & 0 \\ 0 & 0 & \frac{5}{2} \end{pmatrix}$.

∎

Flop-Count. Algorithm [A5.2.2] 5.1 requires roughly $\boldsymbol{\frac{2n^3}{3}}$ **flops**. This can be seen as follows:

- **Step 1**: We compute $(n-1)$ multipliers and update $(n-1)^2$ entries of A. Each multiplier requires one flop and updating each entry requires 2 flops. Thus, Step 1 requires $2(n-1)^2 + (n-1)$ flops.

- **Step 2**: Computing $(n-2)$ multipliers and updating $(n-2)^2$ entries require $2(n-2)^2 + (n-2)$ flops.

In general,

- **Step** k requires $2(n-k)^2 + (n-k)$ flops.

Since there are $(n-1)$ steps, we have

$$\begin{aligned} \text{Total flops} &= \sum_{k=1}^{n-1} 2(n-k)^2 + \sum_{k=1}^{n-1}(n-k) \\ &= 2\frac{n(n-1)(2n-1)}{6} + \frac{n(n-1)}{2} \simeq \left[\frac{2n^3}{3} + O(n^2)\right]. \end{aligned}$$

Recall

- $1^2 + 2^2 + \cdots + r^2 = \frac{r(r+1)(2r+1)}{6}$
- $1 + 2 + \cdots + r = \frac{r(r+1)}{2}$

Gaussian Elimination for a Rectangular Matrix

The above described Gaussian elimination process for an $n \times n$ matrix A can be easily extended to an $m \times n$ matrix to compute its LU factorization, when it exists. The process is identical. However, the number of required steps in this case is $k = \min\{m-1, n\}$. We illustrate this with an example.

MATCOM Note: Algorithm A5.2.2 5.1 has been implemented in MATCOM program **LUGSEL**.

E1.4.2a **Example 5.10** Let

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{pmatrix} \quad m = 3, \quad n = 2.$$

$k = \min(2, 2) = 2.$

k=1. (Eliminate the entries in the first column of A below the diagonal). The multipliers are $m_{21} = -3,\ m_{31} = -5$.

$$\textbf{Update:}\ a_{22} \equiv a_{22}^{(1)} = a_{22} + m_{21}a_{12} = -2$$
$$a_{32} \equiv a_{32}^{(1)} = a_{32} + m_{31}a_{12} = -4$$
$$A \equiv A^{(1)} = \begin{pmatrix} 1 & 2 \\ 0 & -2 \\ 0 & -4 \end{pmatrix}.$$

k=2. (Eliminate the entries in the second column of $A^{(1)}$ below the diagonal). The multiplier is $m_{32} = -2, a_{32} \equiv a_{32}^{(2)} = 0$.

$$\textbf{Updated}\quad A \quad \equiv \quad A^{(2)} = \begin{pmatrix} 1 & 2 \\ 0 & -2 \\ 0 & 0 \end{pmatrix}$$
$$\text{So,}\quad U \quad = \quad \begin{pmatrix} 1 & 2 \\ 0 & -2 \\ 0 & 0 \end{pmatrix}.$$

Note that U in this case is an **3×2 upper triangular matrix.**
Form

$$L = \begin{pmatrix} 1 & 0 & 0 \\ -m_{21} & 1 & 0 \\ -m_{31} & -m_{32} & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 3 & 1 & 0 \\ 5 & 2 & 1 \end{pmatrix}.$$

Verify that

$$LU = \begin{pmatrix} 1 & 0 & 0 \\ 3 & 1 & 0 \\ 5 & 2 & 1 \end{pmatrix} \begin{pmatrix} 1 & 2 \\ 0 & -2 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{pmatrix} = A.$$

■

Flop-Count For an $m \times n$ matrix, Gaussian elimination process requires $mn^2 - \frac{n^3}{3}$ flops [(**Exercise 5.16(a)**)].

MATCOM Note: Algorithm A5.2.2 **5.1** has been implemented in MATCOM program **LUGSEL**.

Difficulties of Gaussian Elimination without Pivoting

As we have seen before, Gaussian elimination without pivoting fails if any of the pivots is zero. However, it is worse yet if any pivot becomes close to zero: *in this case the method can be carried to completion, but the obtained results may be totally wrong.*

Consider the following celebrated example from Forsythe and Moler (1977, pp. 34):

Let Gaussian elimination without pivoting be applied to

$$A = \begin{pmatrix} 0.0001 & 1 \\ 1 & 1 \end{pmatrix},$$

and use three-digit arithmetic. **There is only one step.** We have just one multiplier: $m_{21} = \frac{-1}{10^{-4}} = -10^4$.

$$U = A^{(1)} = \begin{pmatrix} 0.0001 & 1 \\ 0 & 1-10^4 \end{pmatrix} \equiv \begin{pmatrix} 0.0001 & 1 \\ 0 & -10^4 \end{pmatrix}, \text{ and } L = \begin{pmatrix} 1 & 0 \\ 10^4 & 1 \end{pmatrix}$$

The product of the computed L and U is: $LU = \begin{pmatrix} 0.0001 & 1 \\ 2 & 0 \end{pmatrix}$, *which is different from A. Who is to blame?*

Note that the pivot $a_{11}^{(1)} = 0.0001$ is very close to zero (in three-digit arithmetic). **This small pivot gave a large multiplier.** The large multiplier, when used to update the entries, eliminated the smaller entries (e.g., $(1 - 10^4)$ became -10^4).

Fortunately, *we can avoid this small pivot just by row interchanges.* Consider the matrix with the first and second rows interchanged, giving:

$$A' = \begin{pmatrix} 1 & 1 \\ 0.0001 & 1 \end{pmatrix}.$$

Gaussian elimination applied to A' now gives

$$U = A^{(1)} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, \quad L = \begin{pmatrix} 1 & 0 \\ 0.0001 & 1 \end{pmatrix}$$

Note that the pivot in this case is $a_{11}^{(1)} = 1$. The product $LU = \begin{pmatrix} 1 & 1 \\ 0.0001 & 1.0001 \end{pmatrix} = A'$.

Remark: It is true that with the interchange above, we now obtained an LU factorization of the matrix A', a permuted version of the matrix A, and not of the original matrix. However, as we will see in **Chapter 6**, this will suffice our purpose for solving a linear system of equations.

5.2.3 Permutation Matrices and their Properties

A nonzero square matrix P is called a **permutation matrix** if there is exactly one nonzero entry in each row and column which is 1 and the rest are all zero. Thus, if $(\alpha_1, \ldots, \alpha_n)$ is a permutation of $(1, 2, \ldots, n)$, then the associated permutation matrix P is given by

$$P = \begin{pmatrix} e_{\alpha_1}^T \\ \vdots \\ e_{\alpha_n}^T \end{pmatrix},$$

where e_i^T is the i^{th} row of the $n \times n$ identity matrix I. Similarly,

$$P = (e_{\alpha_1}, e_{\alpha_2}, \ldots, e_{\alpha_n}),$$

where e_i is the i^{th} column of I, is a permutation matrix.

E1.4.1 **Example 5.11**

$$P_1 = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}, \quad P_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad P_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

are all permutation matrices. ∎

Effects of Pre-multiplication and Post-multiplication by a Permutation Matrix.

$$\text{If } P_1 = \begin{pmatrix} e_{\alpha_1^T} \\ \vdots \\ e_{\alpha_n^T} \end{pmatrix}, \text{ then } P_1 A = \begin{pmatrix} \alpha_1\text{th row of } A \\ \alpha_2\text{th row of } A \\ \vdots \\ \alpha_n\text{th row of } A \end{pmatrix}.$$

Similarly, if $P_2 = (e_{\alpha_1} e_{\alpha_1} \cdots e_{\alpha_n})$, where e_{α_i} is the i^{th} column of A, then $AP_2 =$ (α_1th column of A, α_2th column of A, …, α_nth column of A).

Thus, the effect of premultiplication of A by a permutation matrix is permutation of the associated rows of A, and that of postmultiplication is the permutation of the associated columns.

E1.4.2 **Example 5.12**

1. $A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{13} & a_{23} & a_{33} \end{pmatrix}, \ P_1 = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} e_2^T \\ e_3^T \\ e_1^T \end{pmatrix}; P_1 A = \begin{pmatrix} a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \\ a_{11} & a_{12} & a_{13} \end{pmatrix} = \begin{pmatrix} \text{2nd row of } A \\ \text{3rd row of } A \\ \text{1st row of } A \end{pmatrix}$

2. $P_1 = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix} = (e_3, e_1, e_2); \ AP_1 = \begin{pmatrix} a_{13} & a_{11} & a_{12} \\ a_{23} & a_{21} & a_{22} \\ a_{33} & a_{31} & a_{32} \end{pmatrix}$
   $=$ (3rd column of A, 1st column of A, 2nd column of A)

∎

An important property of a permutation matrix P is that **it is orthogonal**, that is, $PP^T = I$. As a consequence of this, we have

- The inverse of a permutation matrix P is its transpose and it is also a permutation matrix.
- The product of two permutation matrices is a permutation matrix, and therefore is orthogonal.

5.2.4 Gaussian Elimination with Partial Pivoting (GEPP)

As the above example suggests, disaster in Gaussian elimination without pivoting can perhaps be avoided by identifying a "good pivot" (*a pivot as large as possible in magnitude*) at each step, before the process of elimination is applied. The good pivot may be located among the entries in a column or among all the entries in a submatrix of the current matrix. In the former case, since the search is only partial, the method is called **partial pivoting**; in the latter case, the method is called **complete pivoting**. *It is important to note that the purpose of pivoting is to prevent large growth in the reduced matrices which can wipe out original data.* One way to do this is to keep multipliers less than or equal to one in magnitude, and this is exactly what is accomplished by pivoting. However, *large multipliers do not necessarily mean instability* (see our discussion of Gaussian elimination without pivoting for symmetric positive definite matrices in **Chapter 6**). We first describe Gaussian elimination with partial pivoting (GEPP).

The process consists of **$(n-1)$ steps**.

The process is just a slight modification of Gaussian elimination in the following sense:

At each step do the following:

- Identify the pivot as the largest entry (in magnitude) among all the entries in the pivot column.
- Interchange the appropriate rows to bring the pivot entry to the diagonal position of the current matrix.
- Perform Gaussian Elimination to the row-permuted matrix.

The process is illustrated with a 4×4 example in the following. *For this example, we assume that rows* 3, 4, *and* 4 *are pivot rows in* Step 1, Step 2, and Step 3, respectively.

Step 1.

$$\to \underset{\textit{Pivot Identification}}{\overset{A}{\begin{pmatrix} a_{11} & a_{12} & a_{13} & a14 \\ a_{21} & a_{22} & a_{23} & a_{24} \\ \boxed{a_{31}} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{pmatrix}}} \to \underset{\textit{Row Interchange (1st and 3rd)}}{\overset{\textit{Permuted } A}{\begin{pmatrix} \boxed{a_{31}} & a_{32} & a_{33} & a_{34} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{11} & a_{12} & a_{13} & a_{14} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{pmatrix}}} \to$$

$$\underset{\textit{Gaussian Elimination}}{\overset{A^{(1)}}{\begin{pmatrix} a_{31} & a_{32} & a_{33} & a_{34} \\ 0 & a_{22}^{(a)} & a_{23}^{(1)} & a_{24}^{(1)} \\ 0 & a_{12}^{(1)} & a_{13}^{(1)} & a_{14}^{(1)} \\ 0 & a_{42}^{(1)} & a_{43}^{(1)} & a_{44}^{(1)} \end{pmatrix}}}$$

Step 2.

$$\underset{\textit{Pivot Identification}}{\overset{A^{(1)}}{\begin{pmatrix} a_{31} & a_{32} & a_{33} & a_{34} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & a_{24}^{(1)} \\ 0 & a_{12}^{(1)} & a_{13}^{(1)} & a_{14}^{(1)} \\ 0 & \boxed{a_{42}^{(1)}} & a_{43}^{(1)} & a_{44}^{(1)} \end{pmatrix}}} \rightarrow \underset{\substack{\textit{Row Interchange} \\ \text{(2nd and 4th)}}}{\overset{\textit{Permuted } A}{\begin{pmatrix} a_{31} & a_{32} & a_{33} & a_{34} \\ 0 & a_{42}^{(1)} & a_{43}^{(1)} & a_{44}^{(1)} \\ 0 & a_{12}^{(1)} & a_{13}^{(1)} & a_{14}^{(1)} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & a_{24}^{(1)} \end{pmatrix}}} \rightarrow$$

$$\underset{\textit{Gaussian Elimination}}{\overset{A^{(2)}}{\begin{pmatrix} a_{31} & a_{32} & a_{33} & a_{34} \\ 0 & a_{42}^{(1)} & a_{43}^{(1)} & a_{44}^{(1)} \\ 0 & 0 & a_{13}^{(2)} & a_{14}^{(2)} \\ 0 & 0 & a_{23}^{(2)} & a_{24}^{(2)} \end{pmatrix}}}$$

Step 3.

$$\underset{\textit{Pivot Identification}}{\overset{A^{(2)}}{\begin{pmatrix} a_{31} & a_{32} & a_{33} & a_{34} \\ 0 & a_{42}^{(1)} & a_{43}^{(1)} & a_{44}^{(1)} \\ 0 & 0 & a_{13}^{(2)} & a_{14}^{(2)} \\ 0 & 0 & \boxed{a_{23}^{(2)}} & a_{24}^{(2)} \end{pmatrix}}} \rightarrow \underset{\substack{\textit{Row Interchange} \\ \text{(3rd and 4th)}}}{\overset{\textit{Permuted } A^{(2)}}{\begin{pmatrix} a_{31} & a_{32} & a_{33} & a_{34} \\ 0 & a_{42}^{(1)} & a_{43}^{(1)} & a_{44}^{(1)} \\ 0 & 0 & a_{23}^{(2)} & a_{24}^{(2)} \\ 0 & 0 & a_{13}^{(2)} & a_{14}^{(2)} \end{pmatrix}}} \rightarrow$$

$$\underset{\textit{Gaussian Elimination}}{\overset{A^{(3)}}{\begin{pmatrix} a_{31} & a_{32} & a_{33} & a_{34} \\ 0 & a_{42}^{(1)} & a_{43}^{(1)} & a_{44}^{(1)} \\ 0 & 0 & a_{23}^{(2)} & a_{24}^{(2)} \\ 0 & 0 & 0 & a_{14}^{(3)} \end{pmatrix}}}$$

General Process:

***k^{th}* Step:** Set $A^{(0)} = A$. Then to obtain the matrix $A^{(k)} = (a_{ij}^{(k)})$ from $A^{(k-1)}$ at the previous step, do as follows:

1. Identify the largest element in magnitude among all the elements of the column k below the row $(k-1)$ of the matrix $A^{(k-1)}$. Let it be $a_{rk,k}^{(k-1)}$.

2. Interchange the rows r_k and k to bring $a_{rk,k}^{(k-1)}$ to the diagonal position.

3. Apply Gaussian elimination without row interchanges with $a_{r_k,k}^{(k-1)}$ as the pivot to the submatrix consisting of the rows k through n and columns k through n.

GEPP in Terms of Matrix Multiplications

Observe that

- Row interchange is equivalent to premultiplying the matrix by a suitable permutation matrix.
- Gaussian elimination is equivalent to premultiplying the matrix by an elementary matrix.

So, we can write

$$\begin{aligned} A^{(1)} &= M_1P_1A; A^{(2)} = M_2P_2A^{(1)} \\ &\vdots \\ A^{(n-1)} &= M_{n-1}p_{n-1}A^{(n-2)} \end{aligned}$$

For $n = 4$, the complete process is:

$$A = \begin{pmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{pmatrix}$$

Step 1. $A \overset{P_1}{\to} P_1A \overset{M_1}{\to} M_1P_1A = \begin{pmatrix} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & \times & \times & \times \end{pmatrix} = A^{(1)}$

Step 2. $A^{(1)} \overset{P_2}{\to} P_2A^{(1)} \overset{M_2}{\to} M_2P_2A^{(1)} = M_2P_2M_1P_1A = \begin{pmatrix} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & 0 & \times & \times \\ 0 & 0 & \times & \times \end{pmatrix} =$
$A^{(2)}$

Step 3. $A^{(2)} \overset{P_3}{\to} P_3A^{(2)} \overset{M_3}{\to} M_3P_3A^{(2)} = M_3P_3M_2P_2M_1P_1A = \begin{pmatrix} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times \end{pmatrix} =$
$A^{(3)}$.

LU Factorization from Gaussian Elimination with Partial Pivoting

We will now show that Gaussian elimination with partial pivoting yields a factorization of A in the form

$$PA = LU,$$

where P *is a permutation matrix,* L *is a unit lower triangular* matrix and U *is an upper triangular matrix*. This will be shown in two steps:

First, It will be shown that Gaussian eliminations with partial pivoting directly yields the factorization: $MA = U$, were M is a permuted elementary matrix and U is an upper triangular matrix.
$\boldsymbol{n = 4}$: Since $A^{(3)}$ is upper triangular, we set $U = A^{(3)}$. Then from Step 3, we have

$$U = A^{(3)} = M_3P_3M_2P_2M_1P_1A = MA$$

where

$$M = M_3P_3M_2P_2M_1P_1$$

For an $n \times n$ matrix:

- $M = M_{n-1}P_{n-1}M_{n2}P_{n-2}\dots M_2P_2M_1P_1; U = A^{(n-1)}$.

Second, It will be shown how to extract the matrices P and L from $MA = U$ factorization, so that we have $PA = LU$.

$\boldsymbol{n = 4}$:

$$\begin{aligned} U &= M_3P_3M_2P_2M_1P_1A \\ &= M_3(P_3M_2P_3)(P_3P_2M_1P_2P_3)(P_3P_2P_1)A(\text{ Note that } P_3^2 = P_2^2 = I) \\ &= M_3'M_2'M_1'PA, \end{aligned}$$

where

$$M_3' = M_3,\ M_2' = P_3M_2P_3,\ M_1' = P_3P_2M_1P_2P_3$$

and

$$P = P_3P_2P_1.$$

So, setting $L = (M_1')^{-1}(M_2')^{-1}M_3')^{-1}$, we have $LU = PA$.

For an $n \times n$ matrix: The matrices P and L are given by:

$$P = P_{n-1}P_{n-2}\dots P_2P_1,$$

$$L = (M_1')^{-1}(M_2')^{-1}\dots(M_{n-1}')^{-1}.$$

Constructing the Matrix L

The matrix L is unit lower triangular and easily computable. Observe that

- Each M_i' is the same as M_i except that the multipliers are now permuted (this is an effect of multiplication by permutation matrices).
- $(M_i')^{-1}$ is the same as M_i' except that the multipliers are now negated.

Thus, *as in the case of Gaussian elimination without pivoting, we see that the matrix L is a unit lower triangular matrix.*

E5.2.6 **Example 5.13** Consider Example E5.2.3 5.6 again, *this time with partial pivoting.*

Step 1. Permuted $A \equiv \begin{pmatrix} 4 & 5 & 6 \\ 2 & 2 & 3 \\ 1 & 2 & 4 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 2 & 2 & 3 \\ 4 & 5 & 6 \\ 1 & 2 & 4 \end{pmatrix} = P_1A$

$$A^{(1)} = \begin{pmatrix} 4 & 5 & 6 \\ 0 & -\frac{1}{2} & 0 \\ 0 & \boxed{\frac{3}{4}} & \frac{5}{2} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ -\frac{1}{2} & 1 & 0 \\ -\frac{1}{4} & 0 & 1 \end{pmatrix} \begin{pmatrix} 4 & 5 & 6 \\ 2 & 2 & 3 \\ 1 & 2 & 4 \end{pmatrix} = M_1P_1A$$

Step 2. Permuted $A^{(1)} \equiv \begin{pmatrix} 4 & 5 & 6 \\ 0 & \frac{3}{4} & \frac{5}{2} \\ 0 & -\frac{1}{2} & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 4 & 5 & 6 \\ 0 & -\frac{1}{2} & 0 \\ 0 & \frac{3}{4} & \frac{5}{2} \end{pmatrix} =$
$P_2 A^{(1)}$

$$A^{(2)} = \begin{pmatrix} 4 & 5 & 6 \\ 0 & \frac{3}{4} & \frac{5}{2} \\ 0 & 0 & \frac{5}{3} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \frac{2}{3} & 1 \end{pmatrix} \begin{pmatrix} 4 & 5 & 6 \\ 0 & \frac{3}{4} & \frac{5}{2} \\ 0 & -\frac{1}{2} & 0 \end{pmatrix} =$$
$$M_2 P_2 A^{(1)} = M_2 P_2 M_1 P_1 A.$$

- **Factorization** $\boldsymbol{MA = U}$

$$M = M_2 P_2 M_1 P_1 = \begin{pmatrix} 0 & 1 & 0 \\ 0 & -\dfrac{1}{4} & 1 \\ 1 & -\dfrac{2}{3} & \dfrac{2}{3} \end{pmatrix} \qquad U = \begin{pmatrix} 4 & 5 & 6 \\ 0 & \dfrac{3}{4} & \dfrac{5}{2} \\ 0 & 0 & \dfrac{5}{3} \end{pmatrix} = A^{(2)}$$

- **Factorization** $\boldsymbol{PA = LU}$

$$P = P_2 P_1 = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}.$$

$$L = (M_1')^{-1}\ (M_2')^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ \dfrac{1}{4} & 1 & 0 \\ \dfrac{1}{2} & -\dfrac{2}{3} & 1 \end{pmatrix}. \quad U = A^{(2)} = \begin{pmatrix} 4 & 5 & 6 \\ 0 & \dfrac{3}{4} & \dfrac{5}{2} \\ 0 & 0 & \dfrac{5}{3} \end{pmatrix}.$$

∎

Storage Scheme for LU Factorization
by Gaussian Elimination with Partial Pivoting

- The multipliers can be stored in the appropriate places of the lower triangular part of A (below the diagonal) as they are computed.
- A can be overwritten with each $A^{(k)}$ as soon as the latter is formed and thus, the final upper triangular matrix $U = A^{(n-1)}$ will be stored in the upper triangular part of A (including the diagonal).
- The permutation indices r_k have to be stored in a separate single subscripted integer array.

In view of our above discussion, we can now formulate the following **practical algorithm** for LU factorization with partial pivoting.

ALGORITHM 5.2. LU Factorization using Gaussian Elimination with Partial Pivoting.

A5.2.3

Input: An $n \times n$ matrix A.
Outputs: **(i)** An upper triangular matrix U, **(ii)** the permutation indices r_k needed to form the permutation matrix P, and **(iii)** the multipliers m_{ik} needed to form the unit lower triangular matrix L. The result is: $PA = LU$.
Storage: The storage arrangements for U and the multipliers are the same as of Algorithm A5.2.2 5.1. The permutation indices are stored in a separate array.

For $k = 1, 2, \ldots, n-1$ do

1. (**Find the pivot row**). Find r_k so that $|a_{r_k,k}| = \max_{k \le i \le n} |a_{ik}|$. Save r_k. If $a_{r_k,k} = 0$, then stop. Otherwise, continue.
2. (**Interchange the rows** r_k **and** k). $a_{kj} \leftrightarrow a_{rk}j$ $(j = k,\ k+1, \ldots, n)$.
3. (**Form the multipliers**). $a_{ik} \equiv m_{ik} = -\dfrac{a_{ik}}{a_{kk}}$ $(i = k+1, \ldots, n)$
4. (**Update the entries**). $a_{ij} \equiv a_{ij} + m_{ik}a_{kj} = a_{ij} + a_{ik}a_{kj}$ $(i = k+1, \ldots, n;\ j = k+1, \ldots, n)$.

End

Flop-count. Algorithm A5.2.3 **5.2** requires about $\mathbf{2\frac{n^3}{3}}$ flops and $O(n^2)$ comparisons. (Note that the search for the pivot at step k requires $(n-k)$ comparisons).

Note: *Algorithm* A5.2.3 *5.2 does not give the matrices L and P explicitly. However, these can be constructed easily as explained above, from the multipliers and the permutation indices, respectively.*

E5.2.7 **Example 5.14** Let

$$A = \begin{pmatrix} 1 & 2 & 4 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}.$$

$\boldsymbol{k = 1}$.

1. **The pivot entry is** 7: $r_1 = 3$.

2. **Interchange rows 3 and** 1:

$$A \equiv \begin{pmatrix} 7 & 8 & 9 \\ 4 & 5 & 6 \\ 1 & 2 & 4 \end{pmatrix}.$$

3. **Form the multipliers:**

$$a_{21} \equiv m_{21} = -\frac{4}{7}, \quad a_{31} \equiv m_{31} = -\frac{1}{7}.$$

4. **Update:**

$$A \equiv \begin{pmatrix} 7 & 8 & 9 \\ 0 & \frac{3}{7} & \frac{6}{7} \\ 0 & \boxed{\frac{6}{7}} & \frac{19}{7} \end{pmatrix}.$$

$\boldsymbol{k = 2}$.

1. **The pivot entry is** $\frac{6}{7}$: $r_2 = 3$**.**

2. **Interchange rows 2 and 3:**

$$A \equiv \begin{pmatrix} 7 & 8 & 9 \\ 0 & \frac{6}{7} & \frac{19}{7} \\ 0 & \frac{3}{7} & \frac{6}{7} \end{pmatrix}.$$

3. **Form the multipliers:**

$$m_{32} = -\frac{1}{2}.$$

4. **Update:**

$$A \equiv U = \begin{pmatrix} 7 & 8 & 9 \\ 0 & \frac{6}{7} & \frac{19}{7} \\ 0 & 0 & -\frac{1}{2} \end{pmatrix}.$$

Form $\boldsymbol{L}$ and $\boldsymbol{P}$:

$$P = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \; L = \begin{pmatrix} 1 & 0 & 0 \\ -m_{31} & 1 & 0 \\ -m_{21} & -m_{32} & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ \frac{1}{7} & 1 & 0 \\ \frac{4}{7} & \frac{1}{2} & 1 \end{pmatrix}$$

∎

MATCOM Note: Algorithm 5.2 has been implemented in MATCOM program **PARPIV**.

5.2.5 Gaussian Elimination with Complete Pivoting (GECP)

In Gaussian elimination with complete pivoting, at the k^{th} step, the search for the pivot is made among all the entries of the submatrix below the first $(k-1)$ rows. Set $A^{(0)} = A$. Thus, to obtain $A^{(k)}$ from $A^{(k-1)}$, $k = 1, \ldots, n$ do the following:

- Identify the largest element in magnitude among all the elements of the submatrix obtained by deleting the first $(k-1)$ rows and $(k-1)$ columns. Let it be $a_{rs}^{(k-1)}$.
- Interchange the rows k and r followed by the interchange of the columns k and s.
- Apply GE without row interchange with $a_{rs}^{(k-1)}$ as the pivot to the submatrix consisting of the rows k through n and columns k through n.

In terms of matrix multiplications, this then means:

$$A^{(k)} = M_k P_k A^{(k-1)} Q_k,$$

where M_k is an elementary matrix and P_k is the permutation matrix obtained by interchanging the rows k and r of the identity matrix. Similarly for the matrix Q_k. The matrix $A^{(k)}$ has zeros on the k^{th} column below the (k,k) entry. The matrix M_k can of course be computed in two smaller steps as before.

At the end of the $(n-1)^{th}$ step, the matrix $A^{(n-1)}$ is an upper triangular matrix.

Obtaining Factorization: $PAQ = LU$.

Set

$$A^{(n-1)} = U. \tag{5.4}$$ eq5.2.4

Define

$$Q = Q_1 \cdots Q_{n-1}, \ P = P_{n-1}, P_{n-2}, \ldots, P_1 \tag{5.5}$$ eq5.2.5

and

$$L = P(M_{n-1}, P_{n-1}, \ldots, M_1 P_1)^{-1}. \tag{5.6}$$ eq5.2.6

Then it can be shown (see Golub and Van Loan (1996) that

$$PAQ = LU,$$

where P and Q are both *permutation matrices* and L is *unit triangular* and U is *upper triangular*.

A Practical Scheme for Gaussian Elimination with Complete Pivoting

Remarks similar to those as in the case of partial pivoting hold. The matrices P_k, Q_k $P_k A^{(k-1)} Q_k$, M_k and $M_k P_k A^{(k-1)} Q_k$ do not have to be formed explicitly wasting storage unnecessarily. It is enough to save the indices and the multipliers.

Here is a practical scheme for **complete pivoting,** which does not show the explicit formation of the matrices $P_k, Q_k, M_k, M_k A$ and $P_k A Q_k$. **Note that partial pivoting is just a special case of complete pivoting.**

A5.2.4

ALGORITHM 5.3. LU Factorization using Gaussian Elimination with Complete Pivoting.

Input: An $n \times n$ matrix A

Outputs: (i) An upper triangular matrix U, (ii) permutation indices r_k and s_k from which permutation matrices P and Q can be formed, and (iii) the multipliers m_{ik} from which the lower triangular matrix L can be constructed. The result is: $PAQ = LU$.

Storage: The storage schemes for U and the multipliers are the same as GEPP. The indices r_k and s_k are saved in separate arrays.

For $k = 1, 2, \ldots, n-1$ do

1. Find **the pivot indices** r_k and s_k such that $|a_{r_k,s_k}| = \max\{|a_{ij}| : i, j \geq k\}$, and **save** r_k and s_k.

   If $a_{r_k,s_k} = 0$, then stop. Otherwise, continue.

2. **(Interchange the rows** r_k **and** k**)** $a_{kj} \leftrightarrow a_{r_k,j}$ $(j = k, k+1, \ldots, n)$.

3. **Interchange the columns** s_k **and** k**)** $a_{ik} \leftrightarrow a_{i,s_k}$ $(i = 1, 2, \ldots, n)$.

4. **(Form the multipliers)** $a_{ik} \equiv m_{ik} = -\dfrac{a_{ik}}{a_{kk}}$ $(i = k+1, \ldots, n)$.

5. **(Update the entries of** A**)** $a_{ij} \equiv a_{ij} + m_{ik}a_{kj} = a_{ij} + a_{ik}a_{kj}$ $(i = k+1, \ldots, n;\ j = k+1, \ldots, n)$.

End

Note: *Algorithm* A5.2.4 5.3 *does not give the matrices L, P, and Q explicitly; they have to be formed, respectively, from the multipliers m_{ik} and the permutation indices r_k and s_k.*

E5.2.4 **Example 5.15** Triangularize

$$A = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 2 & \boxed{3} \\ 1 & 1 & 1 \end{pmatrix}$$

using complete pivoting.

$\boldsymbol{k = 1}$.

1. The pivot entry is $3 : r = 2, s = 3$.

2. and 3. Interchange rows 1 and 3 followed by interchange of columns 2 and 3:

$$A \equiv \begin{pmatrix} \boxed{3} & 2 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix}.$$

4. and 5. Perform Gaussian elimination taking the entry 3 as pivot.

$$A \equiv A^{(1)} = \begin{pmatrix} 3 & 2 & 1 \\ 0 & \frac{1}{3} & -\frac{1}{3} \\ 0 & \frac{1}{3} & \boxed{\frac{2}{3}} \end{pmatrix}, \; m_{21} = -\frac{1}{3}, \; m_{31} = \frac{1}{3}.$$

$\boldsymbol{k = 2}$.

1. The pivot entry is $\frac{2}{3}$: $r = 3, s = 3$.
2. and 3. Interchange rows 2 and 3 followed by interchange of columns 2 and 3:

$$A \equiv \begin{pmatrix} 3 & 2 & 1 \\ 0 & \boxed{\frac{2}{3}} & \frac{1}{3} \\ 0 & -\frac{1}{3} & \frac{1}{3} \end{pmatrix},$$

4. and 5. Perform Gaussian elimination taking the entry $\frac{2}{3}$ as pivot.

$$A \equiv A^{(2)} = \begin{pmatrix} 3 & 2 & 1 \\ 0 & \frac{2}{3} & \frac{1}{3} \\ 0 & 0 & \frac{1}{2} \end{pmatrix} = U; \; m_{32} = \frac{1}{2}.$$

The readers are invited to compute the unit lower triangular matrix L and the permutation matrices P and Q such that $PAQ = LU$; using the formulas (5.5) and (5.6). ■

Flop-count: Algorithm 5.3 requires $\boldsymbol{2\frac{n^3}{3}}$ **flops** and $\boldsymbol{O(n^3)}$ comparisons.

MATCOM Note: Algorithm 5.3 has been implemented in MATCOM program **COMPIV**.

5.2.6 Summary of Gaussian Elimination and LU Factorizations.

Gaussian elimination schemes without pivoting, with partial pivoting and with complete pivoting, when carried out to completion, yield, respectively:

- $A = LU$ **(GE without pivoting)**
- $PA = LU$ **(GEPP)**
- $PAQ = LU$ **(GECP)**.

Here L is *unit lower triangular,* U is *upper triangular*, and P and Q are *permutation matrices.*

5.3 Stability of Gaussian Elimination

We have seen before that the computed matrices L and U obtained by Gaussian elimination without pivoting can be such that the product LU can be completely different from A. *In fact,* $\frac{||A-LU||}{||A||}$ can be **arbitrarily large**. Specifically, the following result can be proved (See Higham(1996, pp. 175 - 176), Demmel (1997, pp. 47-49)).

Theorem 5.16. ***(Round-off Error Bound for GE).***
The computed matrices L and U, obtained by Gaussian elimination without pivoting satisfy:

$$A + E = LU,$$

where

$$||E|| \leq n\mu||L||\,||U||.$$

Since $U = A^{(n-1)}$, the stability of Gaussian elimination is better understood by measuring the *growth of the elements* in the reduced matrices $A^{(k)}$. (Note that although pivoting keeps the multipliers bounded by unity, the elements in the reduced matrices still can grow arbitrarily).

D5-4 **Definition 5.17.** *The* **growth factor** ρ *is the ratio of the largest element (in magnitude) of* $A, A^{(1)}, \ldots, A^{(n-1)}$ *to the largest element (in magnitude) of* A*:*

$$\rho = \frac{\max(\alpha, \alpha_1, \alpha_2, \ldots, \alpha_{n-1})}{\alpha},$$

where $\alpha = \max_{i,j} |a_{ij}|$ *and* $\alpha_k = \max_{i,j} |a_{ij}^{(k)}|$.

Now, if *partial pivoting* is used, then

- $|l_{ij}| \leq 1$ for all $i \geq j$, since these l_{ij} are the multipliers.
- $|u_{ij}| = |a_{ij}^{(i)}| \leq \rho\, max_{i,j}\, |a_{ij}|$

We then have the following error bound with partial pivoting (For a complete proof, see **Chapter** ch13 **13**).

Theorem 5.18. ***(Round-off Error Property for GEPP).*** *The matrices L and U computed by Gaussian elimination with partial pivoting satisfy*

$$LU = A + E,$$

where

$$||E||_\infty \leq n^2\mu\rho||A||_\infty.$$

The question, therefore, arises how large ρ can be? To answer the question, we start with an example.

E5.3.1 **Example 5.19**

$$A = \begin{pmatrix} 0.0001 & 1 \\ 1 & 1 \end{pmatrix}$$

1. Gaussian elimination without pivoting gives

$$A^{(1)} = U = \begin{pmatrix} 0.0001 & 1 \\ 0 & -10^4 \end{pmatrix}$$

$$\max |a_{ij}^{(1)}| = 10^4, \ \max |a_{ij}| = 1$$

$$\rho = \text{the growth factor} = 10^4$$

2. Gaussian elimination with partial pivoting yields

$$A^{(1)} = U = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$$

$$\max |a_{ij}^{(1)}| = 1, \ \max \ |a_{ij}| = 1$$

$$\rho = \text{the growth factor} = 1$$

■

The question next is: **how large the growth factor ρ in each case can be for an arbitrary matrix?** We answer this question in the following.

Growth Factor for Gaussian Elimination with Partial Pivoting

For Gaussian elimination with partial pivoting, $\rho \le 2^{n-1}$ [**Exercise 5.15**]:

ρ can be as big as 2^{n-1}.

Unfortunately, one can construct matrices for which this bound is attained. Consider the following example:

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

That is,

$$a_{ij} = \begin{cases} 1 & \text{for } j = i, n, \\ -1 & \text{for } j < i, \\ 0 & \text{otherwise.} \end{cases} \quad (5.7)$$ eq5.3.1

Wilkinson (1965, p. 212) has shown that the growth factor ρ for this matrix with partial pivoting is 2^{n-1}. To see this, take the special case with $n = 4$.

$$A = \begin{pmatrix} 1 & 0 & 0 & 1 \\ -1 & 1 & 0 & 1 \\ -1 & -1 & 1 & 1 \\ -1 & -1 & -1 & 1 \end{pmatrix}.$$

$$A^{(1)} = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & -1 & 1 & 2 \\ 0 & -1 & -1 & 2 \end{pmatrix}, \ A^{(2)} = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 4 \\ 0 & 0 & -1 & 4 \end{pmatrix}, \ A^{(3)} = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 4 \\ 0 & 0 & 0 & 8 \end{pmatrix}$$

Thus the growth factor

$$\rho = \frac{8}{1} = 2^3 = 2^{4-1}.$$

Remarks: Note that this is not the only matrix for which $\rho = 2^{n-1}$. Higham and Higham (1987) have identified a set of matrices for which $\rho = 2^{n-1}$. The matrix

$$B = \begin{pmatrix} 0.7248 & 0.7510 & 0.5241 & 0.7510 \\ 0.7317 & 0.1889 & 0.0227 & -0.7510 \\ 0.7298 & -0.3756 & 0.1150 & 0.7511 \\ -0.6993 & -0.7444 & 0.6647 & -0.7500 \end{pmatrix} \tag{5.8}$$ eq5.3.2

is such a matrix. See Higham (1993).

Examples of the above type are rare. Indeed, in many practical examples, the elements of the matrices $A^{(k)}$ very often continue to decrease in size. *Thus, though Gaussian elimination with partial pivoting is not unconditionally stable in theory, but in practice it can be considered as a stable algorithm.*

- **Growth Factor for Gaussian Elimination with Complete Pivoting**

For Gaussian elimination with complete pivoting:

$$\rho \leq \{n \cdot 2^1 \cdot 3^{\frac{1}{2}} \cdot 4^{\frac{1}{3}} \cdots n^{\frac{1}{n-1}}\}^{1/2}.$$

This is a slowly growing function of n. Furthermore, in **practice** this bound is never attained. *Indeed, there was an unproven conjecture by Wilkinson* (1965, p. 213) *that the growth factor for complete pivoting was bounded by n for real $n \times n$ matrices.* Later Cryer (1968) conjectured that $\rho \leq n$ with equality holding if and only if A is a **Hadamard matrix**. An $n \times n$ matrix is a Hadamard matrix if its elements are ± 1 and $HH^T = nI$. Unfortunately, this conjecture has recently been settled by Gould (1991) negatively, for an arbitrary matrix A. Gould (1991) exhibited a 13×13 matrix for which Gaussian elimination with complete pivoting gave the growth factor $\rho = 13.0205$. Edelman (1992) also gave a counterexample to this conjecture by discovering a matrix of order 25 for which $\rho = 32.986341$. In spite of these recent results, *Gaussian elimination with complete pivoting is a stable algorithm.*

The conjecture regarding the growth factor ρ with complete pivoting for Hadamard matrices has been further investigated by several mathematicians recently. What seems to be important in settling this conjecture for Hadamard matrices is to determine the pivot structures and values of the minors of Hadamard matrices. Several results have been obtained in this direction. See the recent papers of Day and Patterson (1988), Koukouvinos, Mitrouli and Seberry (2000, 2001, 2007).

- **Growth Factor of Gaussian Elimination without Pivoting**

  For Gaussian elimination without pivoting, ρ *can be arbitrarily large*, except for a few special cases, as we shall see later, such as *symmetric positive definite and strictly diagonally dominant matrices.* Thus *Gaussian elimination without pivoting is, in general, a completely unstable algorithm.*

5.4 Summary and Table of Comparisons

For easy reference we now review the most important aspects of this chapter.

5.4.1 Elementary Lower Triangular matrix

An $n \times n$ matrix M of the form $M = I + me_k^T$, where $m = (0, 0, \ldots, 0, m_{k+1,k}, \ldots, m_{n,k})^T$, is called an elementary lower triangular matrix of type k.

If M is as given above, then $M^{-1} = I - me_k^T$.

5.4.2 LU factorization

A factorization of A in the form $A = LU$, where L is unit lower triangular and U is upper triangular, is called an **LU factorization** of A. An LU factorization of matrix A does not always exist. If the leading principal minors of A are all different from zero, then the LU factorization of A exists and is unique **(Theorem 5.8).**

The LU factorization of a matrix A, when it exists, is achieved using elementary lower triangular matrices. The process is called *Gaussian elimination without row interchanges* or *Gaussian elimination without pivoting* **(Algorithm 5.1)**.

The process is efficient, requiring only $\mathbf{2\frac{n^3}{3}}$ flops, but *is unstable for arbitrary matrices. Its use is not recommended in practice unless A is symmetric positive definite or column diagonally dominant because, in these cases, the growth factors are 1 and less than equal to* 2, *respectively; see* **Chapter 6**. For decomposition of A into LU in a stable way, row interchanges *(Gaussian elimination with partial pivoting)* **(Algorithm 5.2)** or both row and column interchanges *(Gaussian elimination with complete pivoting)***(Algorithm 5.3)** to identify an appropriate pivot at each step will be needed. Gaussian elimination with partial and complete pivoting yield factorizations: $PA = LU$ and $PAQ = LU$, respectively, where P and Q are permutation matrices.

5.4.3 Stability of Gaussian Elimination

Aspects of stability, instability, and practical stability in terms of the growth factors of Gaussian eliminataion scheme and the associated round-off results are given in **Section 5.3**

6. Table of Comparisons.

We now summarize in the following table the efficiency and stability properties of these computations. We assume that A is $n \times n$.

TABLE 5.1

TABLE OF COMPARISONS

PROBLEM	**METHOD**	**FLOP-COUNT** (APPROXIMATE)	**STABILITY**
Factorization: $A = LU$	Gaussian elimination without row interchange	$\frac{2n^3}{3}$	Unstable in general
Factorization: $PA = LU$	Gaussian elimination with partial pivoting	$\frac{2n^3}{3}+$ ($O(n^2)$ comparisons)	Stable in practice
Factorization: $PAQ = U$	Gaussian elimination with complete pivoting	$\frac{2n^3}{3}+$ ($O(n^3)$ comparisons)	Stable

Concluding Remarks: *Gaussian elimination without pivoting is unstable in general, Gaussian elimination with partial pivoting is stable in practice; Gaussian elimination with complete pivoting is stable.*

5.5 Suggestions for Further Reading

The topics covered in this chapter are standard and can be found in any numerical linear algebra text. The books by Golub and Van Loan (1996) and G. W. Stewart (1973,1998) are rich sources of further knowledge in this area. See also Higham (1996).

Exercises on Chapter ch5 5

(Use MATLAB, whenever appropriate and necessary)

5.1 (a) Show that an elementary lower triangular matrix of type k defined by (5.1) has the form

$$M_k = I + me_k^T,$$

where $m = (0, 0, \ldots, 0, m_{k+1,k}, \ldots, m_{n,k})^T$.

(b) Show that the inverse of M_k in (a) is given by

$$M_k^{-1} = I - me_k^T.$$

(c) Show that the elementary matrix M defined by (5.2) is such that Ma, where $a = (a_{11}, a_{21}, \cdots, a_{n1})^T$ is a multiple of e_1.

5.2 (a) Given

$$a = \begin{pmatrix} 0.00001 \\ 1 \end{pmatrix}.$$

Using 3-digit arithmetic, find an elementary matrix M such that Ma is a multiple of e_1.

(b) Using your computations in (a), find the LU factorization of

$$A = \begin{pmatrix} 0.00001 & 1 \\ 1 & 2 \end{pmatrix}$$

(c) Let $\hat{L}$ and $\hat{U}$ be the computed L and U in part (b). Find

$$\frac{\|A - \hat{L}\hat{U}\|_F}{\|A\|_F}.$$

5.3 Show that the pivots $a_{11}, a_{22}^{(1)}, \ldots, a_{nn}^{(n-1)}$ are nonzero if and only if the first $(n-1)$ leading principal minors of A are nonsingular.
Hint: Let A_r denote the rth leading principal minor of A. Then show that

$$\det\ A_r = a_{11}a_{22}^{(1)} \ldots, a_{rr}^{(r-1)}.$$

5.4 Assuming that LU factorization of A exists, prove that

(a) (**LDU Factorization**) A can be written in the form

$$A = LDU_1,$$

where D is diagonal and L and U_1 are unit lower and upper triangular matrices, respectively.

(b) (***LDL^T* Factorization**) If A is symmetric, then

$$A = LDL^T.$$

(c) Using 5.4(b), prove that if A is symmetric and positive definite, then

$$A = HH^T,$$

where H is a lower triangular matrix with positive diagonal entries. (**This is known as the Cholesky decomposition**.)

5.5 Assuming that LU factorization of A exists, develop an algorithm to compute U by rows and L by columns directly from the equation: $A = LU$.

This is known as **Doolittle reduction.**

5.6 Develop an algorithm to compute the factorization $A = LU$, where U is unit upper triangular and L is lower triangular. This is known as **Crout reduction**.
Hint: Derive the algorithm from the equation $A = LU$.

5.7 Compare the Doolittle and Crout reductions with Gaussian elimination without pivoting with respect to flop-count and storage requirements.

5.8 A matrix G of the form

$$G = I - ge_k^T,$$

is called a **Gauss-Jordan** matrix. Show that, given a vector x with the property that $e_k^T x \neq 0$, there exists a Gauss-Jordan matrix G such that

$$Gx \text{ is a multiple of } e_k.$$

Develop an algorithm to construct Gauss-Jordan matrices $G_1, G_2, \ldots, G_n$ successively such that $(G_n G_{n-1}, \ldots, G_2 G_1)A$ is a diagonal matrix. This is known as **Gauss-Jordan reduction**.

Derive conditions under which Gauss-Jordan reduction can be carried to completion.

Give a flop-count for the algorithm and compare it with those of Gaussian elimination, Crout reduction and Doolittle reductions.

5.9 Given

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 5 & 4 \\ 3 & 4 & 5 \end{pmatrix}$$

find LU factorization of A using Gaussian elimination and Doolittle reduction and Crout reduction.

5.10 Apply the Gauss-Jordan reduction to A of **Exercise 5.9**.

5.11 Prove that the matrix L in each of the factorizations $PA = LU$ and $PAQ = LU$, obtained by using Gaussian elimination with partial and complete pivoting, respectively, is unit lower triangular.

5.12 Given $A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 2 & 3 & 4 & 5 \end{pmatrix}$,

find a permutation matrix P, a unit lower triangular matrix L, and an upper triangular matrix U such that $PA = LU$.

5.13 (a) Find permutation matrices P and Q and unit lower and upper triangular matrices L and U such that $PA = LU$ and $PAQ = LU$ for the following matrices.

i. $A = \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{3} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \end{pmatrix}$, ii. $A = \begin{pmatrix} 100 & 99 & 98 \\ 98 & 55 & 11 \\ 0 & 1 & 1 \end{pmatrix}$,

iii. $A = \begin{pmatrix} 1 & 0 & 1 \\ -1 & 1 & 1 \\ -1 & -1 & 1 \end{pmatrix}$, iv. $A = \begin{pmatrix} 0.00003 & 1.566 & 1.234 \\ 1.5660 & 2.000 & 1.018 \\ 1.2340 & 1.018 & -3.000 \end{pmatrix}$,

(b) For each of the matrices in 5.13(a), find M and U such that $MA = U$.

(c) Compute the growth factor in each case.

5.14 (a) Consider the 5×5 **row matrix**

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0.1 \\ 0 & 1 & 0 & 0 & 0.1 \\ 0 & 0 & 1 & 0 & 0.1 \\ 0 & 0 & 0 & 1 & 0.1 \\ 0.1 & 0.1 & 0.1 & 0.1 & 0.1 \end{pmatrix}.$$

Find the LU factorizations using both GE and GEPP. How many flops are needed? How many flops will be needed if A is an $n \times n$ row matrix?

(b) Repeat 5.14(a) with the following permuted row matrix

$$A' = \begin{pmatrix} 1 & 0.1 & 0.1 & 0.1 & 0.1 \\ 0.1 & 1 & 0 & 0 & 0 \\ 0.1 & 0 & 1 & 0 & 0 \\ 0.1 & 0 & 0 & 1 & 0 \\ 0.1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

and compare your answers with those obtained in 5.14(a).

5.15 Prove that the growth factor $\rho \leq 2^{n-1}$ for Gaussian elimination with partial pivoting, applied to an $n \times n$ matrix.

5.16 (a) Formulate algorithms for LU factorization of an $m \times n$ $(m \geq n)$ matrix using GE without and with partial pivoting.

Show that each algorithm requires about $mn^2 - \dfrac{n^3}{3}$ flops.

(b) Apply your algorithms to

$$(i)\ A = \begin{pmatrix} 0.00001 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \text{ and (ii) } \ A = \text{ rand } (5, 2).$$

MATLAB AND MATCOM PROGRAMS AND PROBLEMS ON CHAPTER 5

M5.1 Based on Algorithm 5.1, write a MATLAB program, called **lugewp** to compute L and U such that $A = LU$ and the associated growth factor gf: $[L, U, gf] =$**lugewp** (A).

Test Data:

$$\text{(i) } A = \begin{pmatrix} 0.00001 & 1 \\ 1 & 1 \end{pmatrix}, \quad \text{(ii) } A = \begin{pmatrix} 1 & 1 \\ 0.00001 & 1 \end{pmatrix},$$

$$\text{(iii) } A = \begin{pmatrix} 10 & 1 & 1 \\ 1 & 10 & 1 \\ 1 & 1 & 20 \end{pmatrix}, \quad \text{(iv) The matrix } A \text{ in (5.7) with } n = 10.$$

(v) $A = 10 \times 10$ Hilbert matrix, (vi) The matrix A in (5.8)

Print (i) $\dfrac{||L||_F||U||_F}{||A||_F}$ and (ii) $\dfrac{||A - LU||_F}{||L||_F||U||_F}$, (iii) $\dfrac{||A - LU||_F}{||A||_F}$, and the growth factor in each case. Write your observations.

M5.2 Based on Algorithm 5.2, write a MATLAB program, called **lugepp** to compute (i) P, L and U such that $PA = LU$, using partial pivoting, and (ii) the associated growth factor gf:

$$[L, U, P, gf] = \textbf{lugepp}(A).$$

Print $||L||_F||U||_F$ and $\dfrac{||PA - LU||_F}{||A||_F}$ and the growth factor for each of the matrices A of Problem M5.1.

M5.3 Based on Algorithm 5.3, write a MATLAB program, called **lugecp** to compute $P, Q, L,$ and U such that $PAQ = LU$, and the associated growth factor gf:

$$[L, U, P, Q, gf] = \textbf{lugecp}(A).$$

Print $||L||_F||U||_F$, $\dfrac{||PAQ - LU||_F}{||A||_F}$, and the growth factor for each of the matrices of Problem M5.1.

M5.4 Write a MATLAB program, called **GSJOR** to implement Gauss-Jordan scheme outlined in Problem 5.8 and apply your program to the matrices of Problem M5.1.

M5.5 (Experiment on the Growth Factor for GEPP). Plot the growth factors for Gaussian elimination with partial pivoting of 100 randomly generated matrices of sizes varying 10, 20, 30, 40, ..., 1000. Write down your observations.

M5.6 *Random triangular matrices usually become more and more ill-conditioned as the dimensions increase.* However, the lower triangular matrices L from LU factorization of a matrix A using GEPP are believed to have low condition numbers. Perform an experiment to verify this statement, as follows: take a random matrix of order 800 and compute its LU factorization using *lugepp* and plot the entries of the inverse of L. Then change the signs of the subdiagonal entries of L randomly to create another lower triangular matrix $\tilde{L}$ and plot the entries of the inverse of $\tilde{L}$. Compute $\max_{i,j} \left|L_{ij}^{-1}\right|$ and $\max_{i,j} \left|\tilde{L}_{ij}^{-1}\right|$. Repeat the above experiment with random matrices with entries uniformly distributed in $[-1, 1]$.