Chapter 4

Gaussian Elimination, LU-Factorization, Cholesky Factorization, Reduced Row Echelon Form

4.1 Motivating Example: Curve Interpolation

Curve interpolation is a problem that arises frequently in computer graphics and in robotics (path planning).

There are many ways of tackling this problem and in this section we will describe a solution using *cubic splines*.

Such splines consist of cubic Bézier curves.

They are often used because they are cheap to implement and give more flexibility than quadratic Bézier curves.

A *cubic Bézier curve* $C(t)$ (in $\mathbb{R}^2$ or $\mathbb{R}^3$) is specified by a list of four *control points* (b_0, b_2, b_2, b_3) and is given parametrically by the equation

$$C(t) = (1-t)^3\, b_0 + 3(1-t)^2 t\, b_1 + 3(1-t)t^2\, b_2 + t^3\, b_3.$$

Clearly, $C(0) = b_0$, $C(1) = b_3$, and for $t \in [0, 1]$, the point $C(t)$ belongs to the convex hull of the control points b_0, b_1, b_2, b_3.

The polynomials

$$(1-t)^3, \quad 3(1-t)^2 t, \quad 3(1-t)t^2, \quad t^3$$

are the *Bernstein polynomials* of degree 3.

Typically, we are only interested in the curve segment corresponding to the values of t in the interval $[0, 1]$.

Still, the placement of the control points drastically affects the shape of the curve segment, which can even have a self-intersection; See Figures 4.1, 4.2, 4.3 illustrating various configuations.

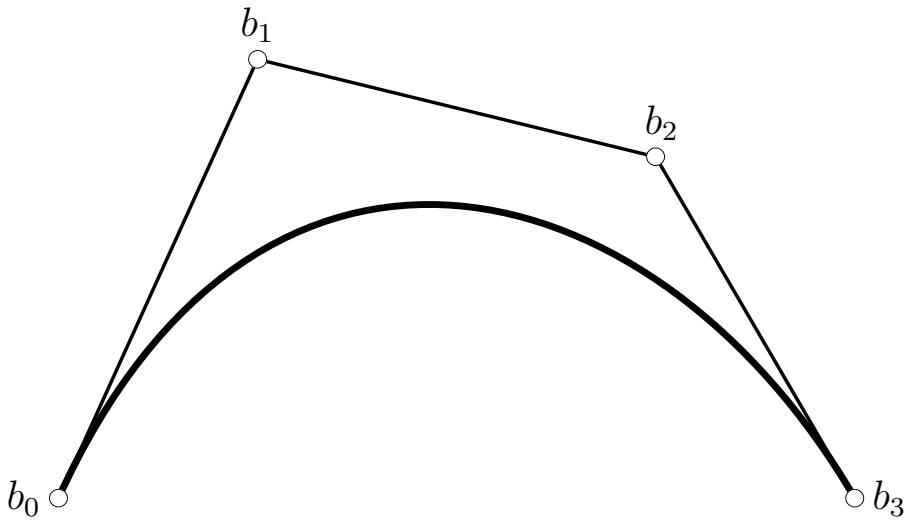

Figure 4.1: A "standard" Bézier curve

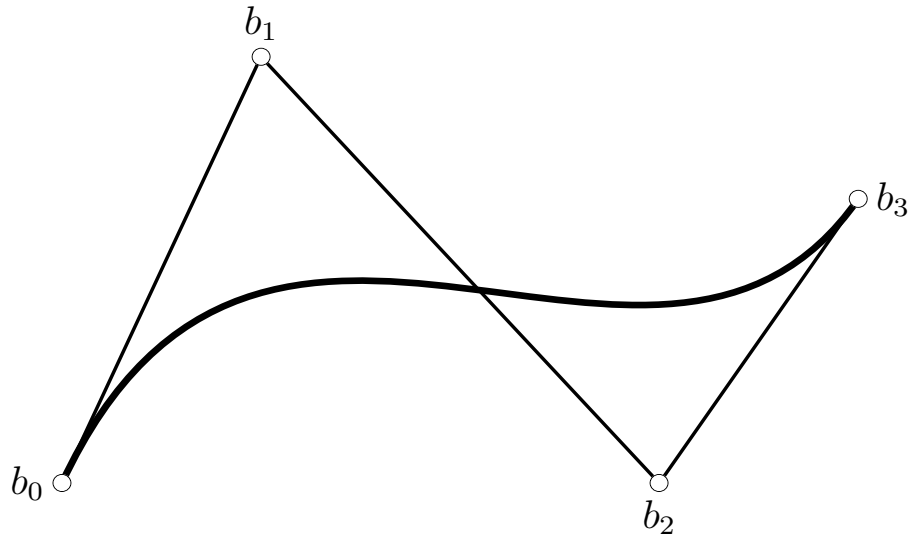

Figure 4.2: A Bézier curve with an inflexion point

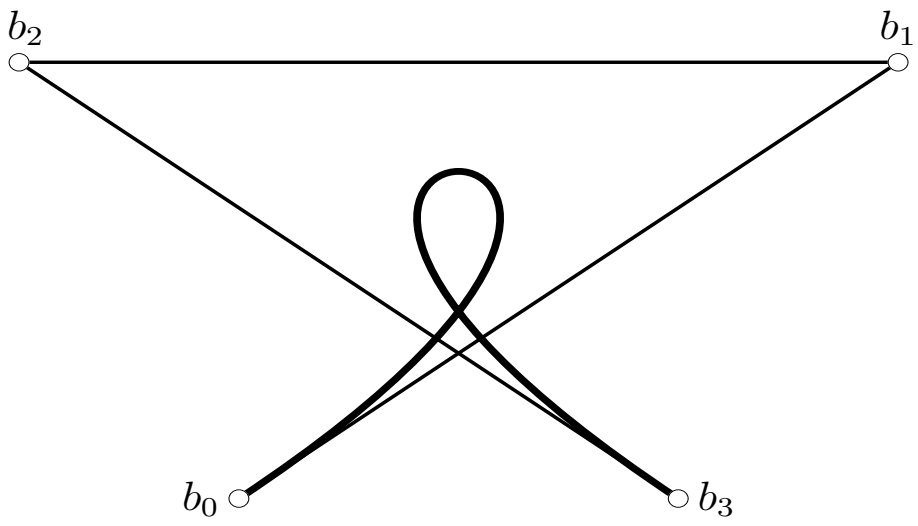

Figure 4.3: A self-intersecting Bézier curve

Interpolation problems require finding curves passing through some given data points and possibly satisfying some extra constraints.

A *Bézier spline curve* F is a curve which is made up of curve segments which are Bézier curves, say $C_1, \ldots, C_m$ $(m \geq 2)$.

We will assume that F defined on $[0, m]$, so that for $i = 1, \ldots, m$,

$$F(t) = C_i(t - i + 1), \quad i - 1 \leq t \leq i.$$

Typically, some smoothness is required between any two junction points, that is, between any two points $C_i(1)$ and $C_{i+1}(0)$, for $i = 1, \ldots, m - 1$.

We require that $C_i(1) = C_{i+1}(0)$ (*C^0-continuity*), and typically that the derivatives of C_i at 1 and of C_{i+1} at 0 agree up to second order derivatives.

This is called *C^2-continuity*, and it ensures that the tangents agree as well as the curvatures.

There are a number of interpolation problems, and we consider one of the most common problems which can be stated as follows:

Problem: Given $N+1$ data points $x_0, \ldots, x_N$, find a C^2 cubic spline curve F, such that $F(i) = x_i$, for all i, $0 \leq i \leq N$ $(N \geq 2)$.

A way to solve this problem is to find $N+3$ auxiliary points $d_{-1}, \ldots, d_{N+1}$ called *de Boor control points* from which N Bézier curves can be found. Actually,

$$d_{-1} = x_0 \quad \text{and} \quad d_{N+1} = x_N$$

so we only need to find $N+1$ points $d_0, \ldots, d_N$.

It turns out that the C^2-continuity constraints on the N Bézier curves yield only $N-1$ equations, so d_0 and d_N can be chosen arbitrarily.

In practice, d_0 and d_N are chosen according to various *end conditions*, such as prescribed velocities at x_0 and x_N. For the time being, we will assume that d_0 and d_N are given.

Figure 4.4 illustrates an interpolation problem involving $N + 1 = 7 + 1 = 8$ data points. The control points d_0 and d_7 were chosen arbitrarily.

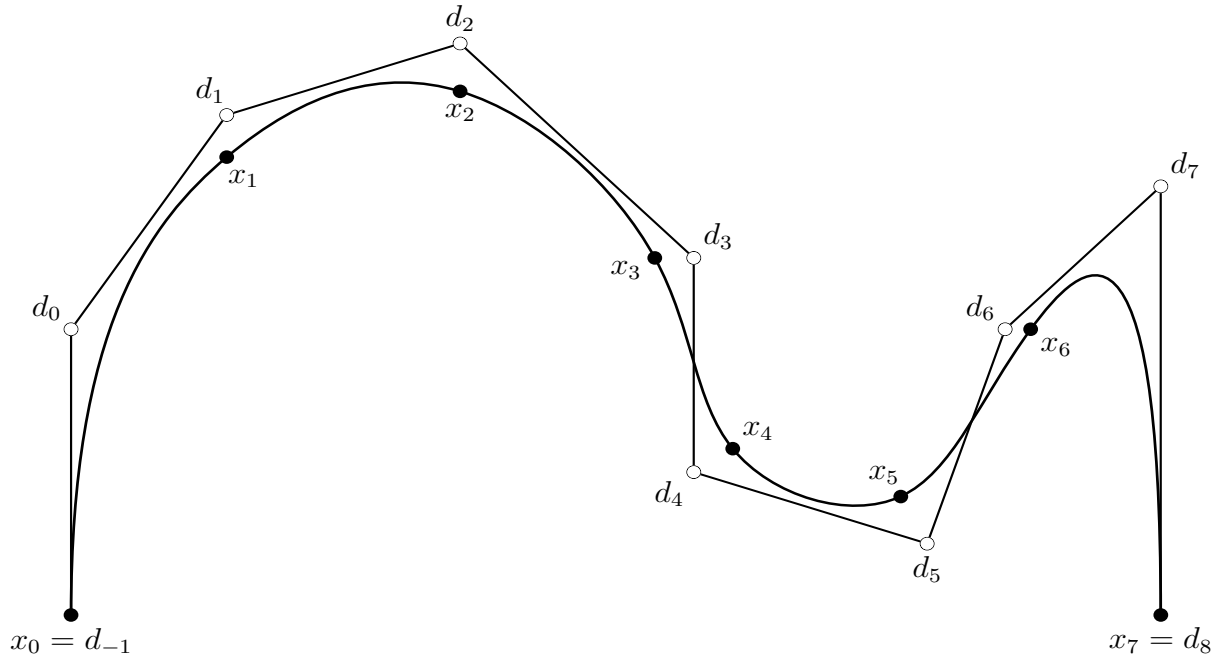

Figure 4.4: A C^2 cubic interpolation spline curve passing through the points $x_0, x_1, x_2, x_3, x_4, x_5, x_6, x_7$

It can be shown that $d_1, \ldots, d_{N-1}$ are given by the linear system

$$\begin{pmatrix} \frac{7}{2} & 1 & & & \\ 1 & 4 & 1 & & 0 \\ & \ddots & \ddots & \ddots & \\ 0 & & 1 & 4 & 1 \\ & & & 1 & \frac{7}{2} \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_{N-2} \\ d_{N-1} \end{pmatrix} = \begin{pmatrix} 6x_1 - \frac{3}{2}d_0 \\ 6x_2 \\ \vdots \\ 6x_{N-2} \\ 6x_{N-1} - \frac{3}{2}d_N \end{pmatrix}.$$

It can be shown that the above matrix is invertible because it is strictly diagonally dominant.

Once the above system is solved, the Bézier cubics $C_1, \ldots, C_N$ are determined as follows (we assume $N \geq 2$):

For $2 \leq i \leq N-1$, the control points $(b_0^i, b_1^i, b_2^i, b_3^i)$ of C_i are given by

$$\begin{aligned} b_0^i &= x_{i-1} \\ b_1^i &= \frac{2}{3}d_{i-1} + \frac{1}{3}d_i \\ b_2^i &= \frac{1}{3}d_{i-1} + \frac{2}{3}d_i \\ b_3^i &= x_i. \end{aligned}$$

The control points $(b_0^1, b_1^1, b_2^1, b_3^1)$ of C_1 are given by

$$\begin{aligned} b_0^1 &= x_0 \\ b_1^1 &= d_0 \\ b_2^1 &= \frac{1}{2}d_0 + \frac{1}{2}d_1 \\ b_3^1 &= x_1, \end{aligned}$$

and the control points $(b_0^N, b_1^N, b_2^N, b_3^N)$ of C_N are given by

$$\begin{aligned} b_0^N &= x_{N-1} \\ b_1^N &= \frac{1}{2}d_{N-1} + \frac{1}{2}d_N \\ b_2^N &= d_N \\ b_3^N &= x_N. \end{aligned}$$

We will now describe various methods for solving linear systems.

Since the matrix of the above system is tridiagonal, there are specialized methods which are more efficient than the general methods. We will discuss a few of these methods.

4.2 Gaussian Elimination and LU-Factorization

Let A be an $n \times n$ matrix, let $b \in \mathbb{R}^n$ be an n-dimensional vector and assume that A is invertible.

Our goal is to solve the system $Ax = b$. Since A is assumed to be invertible, we know that this system has a unique solution, $x = A^{-1}b$.

Experience shows that two counter-intuitive facts are revealed:

(1) One should avoid computing the inverse, A^{-1}, of A explicitly. This is because this would amount to solving the n linear systems, $Au^{(j)} = e_j$, for $j = 1, \ldots, n$, where $e_j = (0, \ldots, 1, \ldots, 0)$ is the jth canonical basis vector of $\mathbb{R}^n$ (with a 1 is the jth slot).

By doing so, we would replace the resolution of a single system by the resolution of n systems, and we would still have to multiply A^{-1} by b.

(2) One does not solve (large) linear systems by computing determinants (using Cramer's formulae).

This is because this method requires a number of additions (resp. multiplications) proportional to $(n+1)!$ (resp. $(n+2)!$).

The key idea on which most direct methods are based is that if A is an *upper-triangular matrix*, which means that $a_{ij} = 0$ for $1 \leq j < i \leq n$ (resp. lower-triangular, which means that $a_{ij} = 0$ for $1 \leq i < j \leq n$), then computing the solution, x, is trivial.

Indeed, say A is an upper-triangular matrix

$$A = \begin{pmatrix} a_{1\,1} & a_{1\,2} & \cdots & a_{1\,n-2} & a_{1\,n-1} & a_{1\,n} \\ 0 & a_{2\,2} & \cdots & a_{2\,n-2} & a_{2\,n-1} & a_{2\,n} \\ 0 & 0 & \ddots & \vdots & \vdots & \vdots \\ & & & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & a_{n-1\,n-1} & a_{n-1\,n} \\ 0 & 0 & \cdots & 0 & 0 & a_{n\,n} \end{pmatrix}.$$

Then, $\det(A) = a_{1\,1}a_{2\,2}\cdots a_{n\,n} \neq 0$, which implies that $a_{i\,i} \neq 0$ for $i = 1, \ldots, n$, and we can solve the system $Ax = b$ from bottom-up by *back-substitution*.

That is, first we compute x_n from the last equation, next plug this value of x_n into the next to the last equation and compute x_{n-1} from it, etc.

This yields

$$\begin{aligned} x_n &= a_{n\,n}^{-1} b_n \\ x_{n-1} &= a_{n-1\,n-1}^{-1}(b_{n-1} - a_{n-1\,n}x_n) \\ &\vdots \\ x_1 &= a_{1\,1}^{-1}(b_1 - a_{1\,2}x_2 - \cdots - a_{1\,n}x_n). \end{aligned}$$

Note that the use of determinants can be avoided to prove that if A is invertible then $a_{i\,i} \neq 0$ for $i = 1, \ldots, n$.

Indeed, it can be shown directly (by induction) that an upper (or lower) triangular matrix is invertible iff all its diagonal entries are nonzero.

If A was lower-triangular, we would solve the system from top-down by *forward-substitution*.

Thus, what we need is a method for transforming a matrix to an equivalent one in upper-triangular form.

This can be done by *elimination*.

Consider the following example:

$$\begin{array}{rcrcrcr} 2x & + & y & + & z & = & 5 \\ 4x & - & 6y & & & = & -2 \\ -2x & + & 7y & + & 2z & = & 9. \end{array}$$

We can eliminate the variable x from the second and the third equation as follows: Subtract twice the first equation from the second and add the first equation to the third. We get the new system

$$\begin{array}{rcrcrcr} 2x & + & y & + & z & = & 5 \\ & - & 8y & - & 2z & = & -12 \\ & & 8y & + & 3z & = & 14. \end{array}$$

This time, we can eliminate the variable y from the third equation by adding the second equation to the third:

$$\begin{array}{rcrcrcr} 2x & + & y & + & z & = & 5 \\ & - & 8y & - & 2z & = & -12 \\ & & & & z & = & 2. \end{array}$$

This last system is upper-triangular.

Using back-substitution, we find the solution: $z = 2$, $y = 1$, $x = 1$.

Observe that we have performed only *row operations*.

The general method is to *iteratively eliminate variables* using simple row operations (namely, adding or subtracting a multiple of a row to another row of the matrix) while simultaneously applying these operations to the vector b, to obtain a system, $MAx = Mb$, where MA is *upper-triangular*.

Such a method is called *Gaussian elimination*.

However, one extra twist is needed for the method to work in all cases: It may be necessary to *permute rows*, as illustrated by the following example:

$$\begin{aligned} x + y + z &= 1 \\ x + y + 3z &= 1 \\ 2x + 5y + 8z &= 1. \end{aligned}$$

In order to eliminate x from the second and third row, we subtract the first row from the second and we subtract twice the first row from the third:

$$\begin{aligned} x + y + z &= 1 \\ 2z &= 0 \\ 3y + 6z &= -1. \end{aligned}$$

Now, the trouble is that y does not occur in the second row; so, we can't eliminate y from the third row by adding or subtracting a multiple of the second row to it.

The remedy is simple: *permute* the second and the third row! We get the system:

$$\begin{aligned} x + y + z &= 1 \\ 3y + 6z &= -1 \\ 2z &= 0, \end{aligned}$$

which is already in triangular form.

Another example where some permutations are needed is:

$$\begin{aligned} z &= 1 \\ -2x + 7y + 2z &= 1 \\ 4x - 6y &= -1. \end{aligned}$$

First, we *permute* the first and the second row, obtaining

$$\begin{aligned} -2x + 7y + 2z &= 1 \\ z &= 1 \\ 4x - 6y &= -1, \end{aligned}$$

and then, we add twice the first row to the third (to eliminate x) obtaining:

$$\begin{aligned} -2x + 7y + 2z &= 1 \\ z &= 1 \\ 8y + 4z &= 1. \end{aligned}$$

Again, we permute the second and the third row, getting

$$\begin{aligned} -2x + 7y + 2z &= 1 \\ 8y + 4z &= 1 \\ z &= 1, \end{aligned}$$

an upper-triangular system.

Of course, in this example, z is already solved and we could have eliminated it first, but for the general method, we need to proceed in a systematic fashion.

We now describe the method of *Gaussian Elimination* applied to a linear system, $Ax = b$, where A is assumed to be invertible.

We use the variable k to keep track of the stages of elimination. Initially, $k = 1$.

(1) The first step is to *pick some nonzero entry, $a_{i\,1}$, in the first column of A*. Such an entry must exist, since A is invertible (otherwise, the first column of A would be the zero vector, and the columns of A would not be linearly independent).

The actual choice of such an element has some impact on the numerical stability of the method, but this will be examined later. For the time being, we assume that some arbitrary choice is made. This chosen element is called the *pivot* of the elimination step and is denoted π_1 (so, in this first step, $\pi_1 = a_{i\,1}$).

(2) Next, we *permute* the row (i) corresponding to the pivot with the first row. Such a step is called *pivoting*. So, after this permutation, the first element of the first row is nonzero.

(3) We now *eliminate* the variable x_1 from all rows except the first by adding suitable multiples of the first row to these rows. More precisely we add $-a_{i\,1}/\pi_1$ times the first row to the ith row, for $i = 2, \ldots, n$. At the end of this step, all entries in the first column are zero except the first.

(4) Increment k by 1. If $k = n$, stop. Otherwise, $k < n$, and then iteratively *repeat* steps (1), (2), (3) on the $(n - k + 1) \times (n - k + 1)$ subsystem obtained by deleting the first $k - 1$ rows and $k - 1$ columns from the current system.

If we let $A_1 = A$ and $A_k = (a_{i\,j}^{(k)})$ be the matrix obtained after $k - 1$ elimination steps $(2 \leq k \leq n)$, then the kth elimination step is applied to the matrix A_k of the form

$$A_k = \begin{pmatrix} a_{1\,1}^{(k)} & a_{1\,2}^{(k)} & \cdots & \cdots & \cdots & a_{1\,n}^{(k)} \\ & a_{2\,2}^{(k)} & \cdots & \cdots & \cdots & a_{2\,n}^{(k)} \\ & & \ddots & \vdots & & \vdots \\ & & & a_{k\,k}^{(k)} & \cdots & a_{k\,n}^{(k)} \\ & & & \vdots & & \vdots \\ & & & a_{n\,k}^{(k)} & \cdots & a_{n\,n}^{(k)} \end{pmatrix}.$$

Actually, note

$$a_{i\,j}^{(k)} = a_{i\,j}^{(i)}$$

for all i, j with $1 \leq i \leq k - 2$ and $i \leq j \leq n$, since the first $k-1$ rows remain unchanged after the $(k-1)$th step.

We will prove later that $\det(A_k) = \pm\det(A)$. Consequently, A_k is invertible.

The fact that A_k is invertible iff A is invertible can also be shown without determinants from the fact that there is some invertible matrix M_k such that $A_k = M_k A$, as we will see shortly.

Since A_k is invertible, some entry $a_{i\,k}^{(k)}$ with $k \leq i \leq n$ is nonzero. Otherwise, the last $n - k + 1$ entries in the first k columns of A_k would be zero, and the first k columns of A_k would yield k vectors in $\mathbb{R}^{k-1}$.

But then, the first k columns of A_k would be linearly dependent and A_k would not be invertible, a contradiction.

So, one the entries $a_{i\,k}^{(k)}$ with $k \leq i \leq n$ can be chosen as pivot, and we permute the kth row with the ith row, obtaining the matrix $\alpha^{(k)} = (\alpha_{j\,l}^{(k)})$.

The new pivot is $\pi_k = \alpha_{k\,k}^{(k)}$, and we zero the entries $i = k+1, \ldots, n$ in column k by adding $-\alpha_{i\,k}^{(k)}/\pi_k$ times row k to row i. At the end of this step, we have A_{k+1}.

Observe that the first $k-1$ rows of A_k are identical to the first $k-1$ rows of A_{k+1}.

The process of Gaussian elimination is illustrated in schematic form below:

$$\begin{pmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{pmatrix} \Longrightarrow \begin{pmatrix} \times & \times & \times & \times \\ \mathbf{0} & \times & \times & \times \\ \mathbf{0} & \times & \times & \times \\ \mathbf{0} & \times & \times & \times \end{pmatrix} \Longrightarrow$$

$$\begin{pmatrix} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & \mathbf{0} & \times & \times \\ 0 & \mathbf{0} & \times & \times \end{pmatrix} \Longrightarrow \begin{pmatrix} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & 0 & \times & \times \\ 0 & 0 & \mathbf{0} & \times \end{pmatrix}.$$

It is easy to figure out what kind of matrices perform the elementary row operations used during Gaussian elimination.

The key point is that if $A = PB$, where A, B are $m \times n$ matrices and P is a square matrix of dimension m, if (as usual) we denote the rows of A and B by $A_1, \ldots, A_m$ and $B_1, \ldots, B_m$, then the formula

$$a_{ij} = \sum_{k=1}^{m} p_{ik} b_{kj}$$

giving the (i, j)th entry in A shows that the *ith row of A is a linear combination of the rows of B*:

$$A_i = p_{i1} B_1 + \cdots + p_{im} B_m.$$

Therefore, *multiplication of a matrix on the left by a square matrix performs row operations*.

Similarly, multiplication of a matrix on the right by a square matrix performs column operations

The permutation of the kth row with the ith row is achieved by multiplying A on the left by the *transposition matrix* $P(i,k)$, which is the matrix obtained from the identity matrix by permuting rows i and k, i.e.,

$$P(i,k) = \begin{pmatrix} 1 & & & & & & & \\ & 1 & & & & & & \\ & & 0 & & & 1 & & \\ & & & 1 & & & & \\ & & & & \ddots & & & \\ & & & & & 1 & & \\ & & 1 & & & & 0 & \\ & & & & & & & 1 \end{pmatrix}.$$

Observe that $\det(P(i,k)) = -1$. Furthermore, $P(i,k)$ is *symmetric* ($P(i,k)^\top = P(i,k)$), and

$$P(i,k)^{-1} = P(i,k).$$

During the permutation step (2), if row k and row i need to be permuted, the matrix A is multiplied on the left by the matrix P_k such that $P_k = P(i,k)$, else we set $P_k = I$.

Adding β times row j to row i is achieved by multiplying A on the left by the *elementary matrix*,

$$E_{i,j;\beta} = I + \beta e_{i\,j},$$

where

$$(e_{i\,j})_{k\,l} = \begin{cases} 1 & \text{if } k = i \text{ and } l = j \\ 0 & \text{if } k \neq i \text{ or } l \neq j, \end{cases}$$

i.e.,

$$E_{i,j;\beta} = \begin{pmatrix} 1 & & & & & & \\ & 1 & & & & & \\ & & 1 & & & & \\ & & & \ddots & & & \\ & & & & 1 & & \\ & \beta & & & & 1 & \\ & & & & & & 1 \end{pmatrix} \text{ or } \begin{pmatrix} 1 & & & & & & \\ & 1 & & & & \beta & \\ & & 1 & & & & \\ & & & \ddots & & & \\ & & & & 1 & & \\ & & & & & 1 & \\ & & & & & & 1 \end{pmatrix}.$$

On the left, $i > j$, and on the right, $i < j$. Observe that the inverse of $E_{i,j;\beta} = I + \beta e_{i\,j}$ is
$E_{i,j;-\beta} = I - \beta e_{i\,j}$, and that $\det(E_{i,j;\beta}) = 1$.

Therefore, during step 3 (the elimination step), the matrix A is multiplied on the left by a product, E_k, of matrices of the form $E_{i,k;\beta_{i,k}}$, with $i > k$.

Consequently, we see that

$$A_{k+1} = E_k P_k A_k,$$

and then

$$A_k = E_{k-1} P_{k-1} \cdots E_1 P_1 A.$$

This justifies the claim made earlier, that $A_k = M_k A$ for some invertible matrix M_k; we can pick

$$M_k = E_{k-1} P_{k-1} \cdots E_1 P_1,$$

a product of invertible matrices.

The fact that $\det(P(i,k)) = -1$ and that $\det(E_{i,j;\beta}) = 1$ implies immediately the fact claimed above:

We always have

$$\det(A_k) = \pm \det(A).$$

Furthermore, since

$$A_k = E_{k-1}P_{k-1}\cdots E_1P_1A$$

and since Gaussian elimination stops for $k = n$, the matrix

$$A_n = E_{n-1}P_{n-1}\cdots E_2P_2E_1P_1A$$

is *upper-triangular*.

Also note that if we let

$$M = E_{n-1}P_{n-1}\cdots E_2P_2E_1P_1,$$

then $\det(M) = \pm 1$, and

$$\det(A) = \pm\det(A_n).$$

The matrices $P(i,k)$ and $E_{i,j;\beta}$ are called *elementary matrices*.

Theorem 4.1. *(Gaussian Elimination) Let A be an $n \times n$ matrix (invertible or not). Then there is some invertible matrix, M, so that $U = MA$ is upper-triangular. The pivots are all nonzero iff A is invertible.*

Remark: Obviously, the matrix M can be computed as

$$M = E_{n-1}P_{n-1} \cdots E_2P_2E_1P_1,$$

but this expression is of no use.

Indeed, what we need is M^{-1}; when no permutations are needed, it turns out that M^{-1} can be obtained immediately from the matrices E_k's, in fact, from their inverses, and no multiplications are necessary.

Remark: Instead of looking for an invertible matrix, M, so that MA is upper-triangular, we can look for an invertible matrix, M, so that *MA is a diagonal matrix*.

Only a simple change to Gaussian elimination is needed.

At every stage, k, after the pivot has been found and pivoting been performed, if necessary, in addition to adding suitable multiples of the kth row to the rows *below* row k in order to zero the entries in column k for $i = k+1, \ldots, n$, also add suitable multiples of the kth row to the rows *above* row k in order to zero the entries in column k for $i = 1, \ldots, k-1$.

Such steps are also achieved by multiplying on the left by elementary matrices $E_{i,k;\beta_{i,k}}$, except that $i < k$, so that these matrices are not lower-triangular matrices.

Nevertheless, at the end of the process, we find that $A_n = MA$, is a diagonal matrix.

This method is called the *Gauss-Jordan factorization*. Because it is more expansive than Gaussian elimination, this method is not used much in practice.

However, Gauss-Jordan factorization can be used to compute the inverse of a matrix, A.

It remains to discuss the choice of the pivot, and also conditions that guarantee that no permutations are needed during the Gaussian elimination process.

We begin by stating a necessary and sufficient condition for an invertible matrix to have an LU-factorization (i.e., Gaussian elimination does not require pivoting).

We say that an invertible matrix, A, has an *LU-factorization* if it can be written as $A = LU$, where U is *upper-triangular* invertible and L is *lower-triangular*, with $L_{i\,i} = 1$ for $i = 1, \ldots, n$.

A lower-triangular matrix with diagonal entries equal to 1 is called a *unit lower-triangular* matrix.

Given an $n \times n$ matrix, $A = (a_{i\,j})$, for any k, with $1 \leq k \leq n$, let $A[1..k, 1..k]$ denote the submatrix of A whose entries are $a_{i\,j}$, where $1 \leq i, j \leq k$.

Proposition 4.2. *Let A be an invertible $n \times n$-matrix. Then, A, has an LU-factorization, $A = LU$, iff every matrix $A[1..k, 1..k]$ is invertible for $k = 1, \ldots, n$. Furthermore, when A has an LU-factorization, we have*

$$\det(A[1..k, 1..k]) = \pi_1 \pi_2 \cdots \pi_k, \quad k = 1, \ldots, n,$$

where π_k is the pivot obtained after $k - 1$ elimination steps. Therefore, the kth pivot is given by

$$\pi_k = \begin{cases} a_{11} = \det(A[1..1, 1..1]) & \text{if } k = 1 \\ \dfrac{\det(A[1..k, 1..k])}{\det(A[1..k-1, 1..k-1])} & \text{if } k = 2, \ldots, n. \end{cases}$$

Corollary 4.3. *(LU-Factorization) Let A be an invertible $n \times n$-matrix. If every matrix $A[1..k, 1..k]$ is invertible for $k = 1, \ldots, n$, then Gaussian elimination requires no pivoting and yields an LU-factorization, $A = LU$.*

The reader should verify that the example below is indeed an LU-factorization.

$$\begin{pmatrix} 2 & 1 & 1 & 0 \\ 4 & 3 & 3 & 1 \\ 8 & 7 & 9 & 5 \\ 6 & 7 & 9 & 8 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 4 & 3 & 1 & 0 \\ 3 & 4 & 1 & 1 \end{pmatrix} \begin{pmatrix} 2 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 2 & 2 \\ 0 & 0 & 0 & 2 \end{pmatrix}.$$

One of the main reasons why the existence of an LU-factorization for a matrix, A, is interesting is that if we need to solve *several* linear systems, $Ax = b$, corresponding to the same matrix, A, we can do this cheaply by solving the two triangular systems

$$Lw = b, \quad \text{and} \quad Ux = w.$$

As we will see a bit later, symmetric positive definite matrices satisfy the condition of Proposition 4.2.

Therefore, linear systems involving symmetric positive definite matrices can be solved by Gaussian elimination without pivoting.

Actually, it is possible to do better: This is the Cholesky factorization.

There is a certain asymmetry in the LU-decomposition $A = LU$ of an invertible matrix A. Indeed, the diagonal entries of L are all 1, but this is generally false for U.

This asymmetry can be eliminated as follows: if

$$D = \mathrm{diag}(u_{11}, u_{22}, \ldots, u_{nn})$$

is the diagonal matrix consisting of the diagonal entries in U (the pivots), then we if let $U' = D^{-1}U$, we can write

$$A = LDU',$$

where L is lower- triangular, U' is upper-triangular, all diagonal entries of both L and U' are 1, and D is a diagonal matrix of pivots.

Such a decomposition is called an *LDU-factorization*.

We will see shortly than if A is symmetric, then $U' = L^\top$.

If a square invertible matrix A has an LU-factorization, then it is possible to find L and U while performing Gaussian elimination.

Recall that at step k, we pick a pivot $\pi_k = a_{ik}^{(k)} \neq 0$ in the portion consisting of the entries of index $j \geq k$ of the k-th column of the matrix A_k obtained so far, we swap rows i and k if necessary (the pivoting step), and then we zero the entries of index $j = k+1, \ldots, n$ in column k.

Schematically, we have the following steps:

$$\begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & a_{ik}^{(k)} & \times & \times & \times \\ 0 & \times & \times & \times & \times \end{pmatrix} \overset{\text{pivot}}{\Longrightarrow} \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & a_{ik}^{(k)} & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \end{pmatrix}$$

$$\overset{\text{elim}}{\Longrightarrow} \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \mathbf{0} & \times & \times & \times \\ 0 & \mathbf{0} & \times & \times & \times \\ 0 & \mathbf{0} & \times & \times & \times \end{pmatrix}.$$

More precisely, after permuting row k and row i (the pivoting step), if the entries in column k below row k are $\alpha_{k+1k}, \ldots, \alpha_{nk}$, then we add $-\alpha_{jk}/\pi_k$ times row k to row j; this process is illustrated below:

$$\begin{pmatrix} a_{kk}^{(k)} \\ a_{k+1k}^{(k)} \\ \vdots \\ a_{ik}^{(k)} \\ \vdots \\ a_{nk}^{(k)} \end{pmatrix} \overset{\text{pivot}}{\Longrightarrow} \begin{pmatrix} a_{ik}^{(k)} \\ a_{k+1k}^{(k)} \\ \vdots \\ a_{kk}^{(k)} \\ \vdots \\ a_{nk}^{(k)} \end{pmatrix} = \begin{pmatrix} \pi_k \\ \alpha_{k+1k} \\ \vdots \\ \alpha_{ik} \\ \vdots \\ \alpha_{nk} \end{pmatrix} \overset{\text{elim}}{\Longrightarrow} \begin{pmatrix} \pi_k \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{pmatrix}.$$

Then, if we write $\ell_{jk} = \alpha_{jk}/\pi_k$ for $j = k+1, \ldots, n$, the kth column of L is

$$\begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ \ell_{k+1k} \\ \vdots \\ \ell_{nk} \end{pmatrix}.$$

Observe that the signs of the multipliers $-\alpha_{jk}/\pi_k$ have been flipped. Thus, we obtain the unit lower triangular matrix

$$L = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ \ell_{21} & 1 & 0 & \cdots & 0 \\ \ell_{31} & \ell_{32} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ \ell_{n1} & \ell_{n2} & \ell_{n3} & \cdots & 1 \end{pmatrix}.$$

It is easy to see (and this is proved in Theorem 4.5) that the inverse of L is obtained from L by flipping the signs of the ℓ_{ij}:

$$L^{-1} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ -\ell_{21} & 1 & 0 & \cdots & 0 \\ -\ell_{31} & -\ell_{32} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ -\ell_{n1} & -\ell_{n2} & -\ell_{n3} & \cdots & 1 \end{pmatrix}.$$

Furthermore, if the result of Gaussian elimination (without pivoting) is $U = E_{n-1} \cdots E_1 A$, then

$$E_k = \begin{pmatrix} 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 1 & 0 & \cdots & 0 \\ 0 & \cdots & -\ell_{k+1k} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & -\ell_{nk} & 0 & \cdots & 1 \end{pmatrix}$$

and

$$E_k^{-1} = \begin{pmatrix} 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 1 & 0 & \cdots & 0 \\ 0 & \cdots & \ell_{k+1k} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \ell_{nk} & 0 & \cdots & 1 \end{pmatrix},$$

so the kth column of E_k is the kth column of L^{-1}.

Here is an example illustrating the method. Given

$$A = A_1 = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & -1 & 0 & 1 \\ 1 & 1 & -1 & 0 \\ 1 & -1 & 0 & -1 \end{pmatrix},$$

we have the following sequence of steps: The first pivot is $\pi_1 = 1$ in row 1, and we substract row 1 from rows 2, 3, and 4. We get

$$A_2 = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & -2 & -1 & 1 \\ 0 & 0 & -2 & 0 \\ 0 & -2 & -1 & -1 \end{pmatrix} \quad L_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}.$$

The next pivot is $\pi_2 = -2$ in row 2, and we subtract row 2 from row 4 (and add 0 times row 2 to row 3). We get

$$A_3 = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & -2 & -1 & 1 \\ 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & -2 \end{pmatrix} \quad L_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}.$$

The next pivot is $\pi_3 = -2$ in row 3, and since the fourth

entry in column 3 is already a zero, we add 0 times row 3 to row 4. We get

$$A_4 = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & -2 & -1 & 1 \\ 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & -2 \end{pmatrix} \qquad L_3 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}.$$

The procedure is finished, and we have

$$L = L_4 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix} \qquad U = A_4 = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & -2 & -1 & 1 \\ 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & -2 \end{pmatrix}.$$

It is easy to check that indeed

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & -2 & -1 & 1 \\ 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & -2 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & -1 & 0 & 1 \\ 1 & 1 & -1 & 0 \\ 1 & -1 & 0 & -1 \end{pmatrix},$$

namely $A = LU$.

The following easy proposition shows that, in principle, A can be premultiplied by some permutation matrix, P, so that PA can be converted to upper-triangular form without using any pivoting.

A *permutation matrix* is a square matrix that has a single 1 in every row and every column and zeros everywhere else.

It is shown in Section 5.1 that every permutation matrix is a product of transposition matrices (the $P(i, k)$s), and that P is invertible with inverse $P^\top$.

Proposition 4.4. *Let A be an invertible $n \times n$-matrix. Then, there is some permutation matrix, P, so that $(PA)[1..k, 1..k]$ is invertible for $k = 1, \ldots, n$.*

Remark: One can also prove Proposition 4.4 using a clever reordering of the Gaussian elimination steps suggested by Trefethen and Bau [34] (Lecture 21).

It is remarkable that if pivoting steps are necessary during Gaussian elimination, a very simple modification of the algorithm for finding an LU-factorization yields the matrices L, U, and P, such that $PA = LU$.

To describe this new method, since the diagonal entries of L are 1s, it is convenient to write

$$L = I + \Lambda.$$

Then, in assembling the matrix Λ while performing Gaussian elimination with pivoting, *we make the same transposition on the rows of Λ (really Λ_{k-1}) that we make on the rows of A (really A_k) during a pivoting step involving row k and row i.*

We also assemble P by starting with the identity matrix and *applying to P the same row transpositions that we apply to A and Λ.*

Here is an example illustrating this method. Given

$$A = A_1 = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & -1 & 0 \\ 1 & -1 & 0 & 1 \\ 1 & -1 & 0 & -1 \end{pmatrix},$$

we have the following sequence of steps: We initialize $\Lambda_0 = 0$ and $P_0 = I_4$.

The first pivot is $\pi_1 = 1$ in row 1, and we substract row 1 from rows 2, 3, and 4. We get

$$A_2 = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & -2 & 0 \\ 0 & -2 & -1 & 1 \\ 0 & -2 & -1 & -1 \end{pmatrix} \quad \Lambda_1 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

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

The next pivot is $\pi_2 = -2$ in row 3, so we permute row 2 and 3; we also apply this permutation to Λ and P:

$$A'_3 = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & -2 & -1 & 1 \\ 0 & 0 & -2 & 0 \\ 0 & -2 & -1 & -1 \end{pmatrix} \qquad \Lambda'_2 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

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

Next, we subtract row 2 from row 4 (and add 0 times row 2 to row 3). We get

$$A_3 = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & -2 & -1 & 1 \\ 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & -2 \end{pmatrix} \qquad \Lambda_2 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}$$

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

The next pivot is $\pi_3 = -2$ in row 3, and since the fourth entry in column 3 is already a zero, we add 0 times row 3 to row 4. We get

$$A_4 = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & -2 & -1 & 1 \\ 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & -2 \end{pmatrix} \qquad \Lambda_3 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}$$

$$P_3 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

The procedure is finished, and we have

$$L = \Lambda_4 + I = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix} \quad U = A_4 = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & -2 & -1 & 1 \\ 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & -2 \end{pmatrix}$$

$$P = P_3 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

It is easy to check that indeed

$$LU = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & -2 & -1 & 1 \\ 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & -2 \end{pmatrix}$$

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

and

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

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

Using the idea in the remark before the above example, we can prove the theorem below which shows the correctness of the algorithm for computing P, L and U using a simple adaptation of Gaussian elimination.

We are not aware of a detailed proof of Theorem 4.5 (see below) in the standard texts.

Although Golub and Van Loan [17] state a version of this theorem as their Theorem 3.1.4, they say that "The proof is a messy subscripting argument."

Meyer [27] also provides a sketch of proof (see the end of Section 3.10).

Theorem 4.5. *For every invertible $n \times n$-matrix, A, the following hold:*

(1) There is some permutation matrix, P, some upper-triangular matrix, U, and some unit lower-triangular matrix, L, so that $PA = LU$ (recall, $L_{i\,i} = 1$ for $i = 1, \ldots, n$). Furthermore, if $P = I$, then L and U are unique and they are produced as a result of Gaussian elimination without pivoting.

(2) If $E_{n-1} \ldots E_1 A = U$ is the result of Gaussian elimination without pivoting, write as usual $A_k = E_{k-1} \ldots E_1 A$ (with $A_k = (a_{ij}^{(k)})$), and let $\ell_{ik} = a_{ik}^{(k)}/a_{kk}^{(k)}$, with $1 \leq k \leq n-1$ and $k+1 \leq i \leq n$. Then

$$L = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ \ell_{21} & 1 & 0 & \cdots & 0 \\ \ell_{31} & \ell_{32} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ \ell_{n1} & \ell_{n2} & \ell_{n3} & \cdots & 1 \end{pmatrix},$$

where the kth column of L is the kth column of E_k^{-1}, for $k = 1, \ldots, n-1$.

(3) If $E_{n-1}P_{n-1}\cdots E_1P_1A = U$ *is the result of Gaussian elimination with some pivoting, write* $A_k = E_{k-1}P_{k-1}\cdots E_1P_1A$, *and define* E_j^k, *with* $1 \leq j \leq n-1$ *and* $j \leq k \leq n-1$, *such that, for* $j = 1, \ldots, n-2$,

$$\begin{aligned} E_j^j &= E_j \\ E_j^k &= P_k E_j^{k-1} P_k, \quad \textit{for } k = j+1, \ldots, n-1, \end{aligned}$$

and

$$E_{n-1}^{n-1} = E_{n-1}.$$

Then,

$$\begin{aligned} E_j^k &= P_k P_{k-1}\cdots P_{j+1} E_j P_{j+1}\cdots P_{k-1}P_k \\ U &= E_{n-1}^{n-1}\cdots E_1^{n-1} P_{n-1}\cdots P_1 A, \end{aligned}$$

and if we set

$$\begin{aligned} P &= P_{n-1}\cdots P_1 \\ L &= (E_1^{n-1})^{-1}\cdots(E_{n-1}^{n-1})^{-1}, \end{aligned}$$

then

$$PA = LU.$$

Furthermore,

$$(E_j^k)^{-1} = I + \mathcal{E}_j^k, \quad 1 \leq j \leq n-1,\ j \leq k \leq n-1,$$

where $\mathcal{E}_j^k$ *is a lower triangular matrix of the form*

$$\mathcal{E}_j^{(k)} = \begin{pmatrix} 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \cdots & \ell_{j+1j}^{(k)} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \ell_{nj}^{(k)} & 0 & \cdots & 0 \end{pmatrix},$$

we have

$$E_j^k = I - \mathcal{E}_j^k,$$

and

$$\mathcal{E}_j^k = P_k \mathcal{E}_j^{k-1}, \quad 1 \leq j \leq n-2,\ j+1 \leq k \leq n-1,$$

where $P_k = I$ *or else* $P_k = P(k,i)$ *for some* i *such that* $k+1 \leq i \leq n$*; if* $P_k \neq I$*, this means that* $(E_j^k)^{-1}$ *is obtained from* $(E_j^{k-1})^{-1}$ *by permuting the entries on row* i *and* k *in column* j*.*

Because the matrices $(E_j^k)^{-1}$ *are all lower triangular, the matrix* L *is also lower triangular.*

In order to find L, define lower triangular matrices Λ_k of the form

$$\Lambda_k = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & \cdots & \cdots & 0 \\ \lambda_{21}^{(k)} & 0 & 0 & 0 & 0 & \vdots & \vdots & 0 \\ \lambda_{31}^{(k)} & \lambda_{32}^{(k)} & \ddots & 0 & 0 & \vdots & \vdots & 0 \\ \vdots & \vdots & \ddots & 0 & 0 & \vdots & \vdots & \vdots \\ \lambda_{k+11}^{(k)} & \lambda_{k+12}^{(k)} & \cdots & \lambda_{k+1k}^{(k)} & 0 & \cdots & \cdots & 0 \\ \lambda_{k+21}^{(k)} & \lambda_{k+22}^{(k)} & \cdots & \lambda_{k+2k}^{(k)} & 0 & \ddots & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \lambda_{n1}^{(k)} & \lambda_{n2}^{(k)} & \cdots & \lambda_{nk}^{(k)} & 0 & \cdots & \cdots & 0 \end{pmatrix}$$

to assemble the columns of L iteratively as follows: let

$$(-\ell_{k+1k}^{(k)}, \ldots, -\ell_{nk}^{(k)})$$

be the last $n-k$ elements of the kth column of E_k, and define Λ_k inductively by setting

$$\Lambda_1 = \begin{pmatrix} 0 & 0 & \cdots & 0 \\ \ell_{21}^{(1)} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \ell_{n1}^{(1)} & 0 & \cdots & 0 \end{pmatrix},$$

then for $k = 2, \ldots, n-1$, *define*

$$\Lambda'_k = P_k \Lambda_{k-1},$$

and

$$\Lambda_k = (I + \Lambda'_k)E_k^{-1} - I =$$

$$\begin{pmatrix}
0 & 0 & 0 & 0 & 0 & \cdots & \cdots & 0 \\
\lambda_{21}^{\prime(k-1)} & 0 & 0 & 0 & 0 & \vdots & \vdots & 0 \\
\lambda_{31}^{\prime(k-1)} & \lambda_{32}^{\prime(k-1)} & \ddots & 0 & 0 & \vdots & \vdots & 0 \\
\vdots & \vdots & \ddots & 0 & 0 & \vdots & \vdots & \vdots \\
\lambda_{k1}^{\prime(k-1)} & \lambda_{k2}^{\prime(k-1)} & \cdots & \lambda_{k\,(k-1)}^{\prime(k-1)} & 0 & \cdots & \cdots & 0 \\
\lambda_{k+11}^{\prime(k-1)} & \lambda_{k+12}^{\prime(k-1)} & \cdots & \lambda_{k+1\,(k-1)}^{\prime(k-1)} & \ell_{k+1k}^{(k)} & \ddots & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
\lambda_{n1}^{\prime(k-1)} & \lambda_{n2}^{\prime(k-1)} & \cdots & \lambda_{n\,k-1}^{\prime(k-1)} & \ell_{nk}^{(k)} & \cdots & \cdots & 0
\end{pmatrix},$$

with $P_k = I$ *or* $P_k = P(k, i)$ *for some* $i > k$.

This means that in assembling L, *row* k *and row* i *of* Λ_{k-1} *need to be permuted when a pivoting step permuting row* k *and row* i *of* A_k *is required.*

Then

$$I + \Lambda_k = (E_1^k)^{-1} \cdots (E_k^k)^{-1}$$
$$\Lambda_k = \mathcal{E}_1^k \cdots \mathcal{E}_k^k,$$

for $k = 1, \ldots, n-1$, *and therefore*

$$L = I + \Lambda_{n-1}.$$

We emphasize again that part (3) of Theorem 4.5 shows the *remarkable fact* that *in assembling the matrix L while performing Gaussian elimination with pivoting, the only change to the algorithm is to make the same transposition on the rows of Λ_k that we make on the rows of A (really A_k) during a pivoting step involving row k and row i.*

We can also assemble P by starting with the identity matrix and *applying to P the same row transpositions that we apply to A and Λ*.

Consider the matrix

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

We set $P_0 = I_4$, and we can also set $\Lambda_0 = 0$. The first step is to permute row 1 and row 2, using the pivot 4. We also apply this permutation to P_0:

$$A'_1 = \begin{pmatrix} 4 & 8 & 12 & -8 \\ 1 & 2 & -3 & 4 \\ 2 & 3 & 2 & 1 \\ -3 & -1 & 1 & -4 \end{pmatrix} \quad P_1 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Next, we subtract 1/4 times row 1 from row 2, 1/2 times row 1 from row 3, and add 3/4 times row 1 to row 4, and start assembling Λ:

$$A_2 = \begin{pmatrix} 4 & 8 & 12 & -8 \\ 0 & 0 & -6 & 6 \\ 0 & -1 & -4 & 5 \\ 0 & 5 & 10 & -10 \end{pmatrix} \qquad \Lambda_1 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1/4 & 0 & 0 & 0 \\ 1/2 & 0 & 0 & 0 \\ -3/4 & 0 & 0 & 0 \end{pmatrix}$$

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

Next we permute row 2 and row 4, using the pivot 5. We also apply this permutation to Λ and P:

$$A_3' = \begin{pmatrix} 4 & 8 & 12 & -8 \\ 0 & 5 & 10 & -10 \\ 0 & -1 & -4 & 5 \\ 0 & 0 & -6 & 6 \end{pmatrix} \qquad \Lambda_2' = \begin{pmatrix} 0 & 0 & 0 & 0 \\ -3/4 & 0 & 0 & 0 \\ 1/2 & 0 & 0 & 0 \\ 1/4 & 0 & 0 & 0 \end{pmatrix}$$

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

Next we add 1/5 times row 2 to row 3, and update Λ_2':

$$A_3 = \begin{pmatrix} 4 & 8 & 12 & -8 \\ 0 & 5 & 10 & -10 \\ 0 & 0 & -2 & 3 \\ 0 & 0 & -6 & 6 \end{pmatrix} \quad \Lambda_2 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ -3/4 & 0 & 0 & 0 \\ 1/2 & -1/5 & 0 & 0 \\ 1/4 & 0 & 0 & 0 \end{pmatrix}$$

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

Next we permute row 3 and row 4, using the pivot -6. We also apply this permutation to Λ and P:

$$A_4' = \begin{pmatrix} 4 & 8 & 12 & -8 \\ 0 & 5 & 10 & -10 \\ 0 & 0 & -6 & 6 \\ 0 & 0 & -2 & 3 \end{pmatrix} \quad \Lambda_3' = \begin{pmatrix} 0 & 0 & 0 & 0 \\ -3/4 & 0 & 0 & 0 \\ 1/4 & 0 & 0 & 0 \\ 1/2 & -1/5 & 0 & 0 \end{pmatrix}$$

$$P_3 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

Finally, we subtract 1/3 times row 3 from row 4, and update Λ_3':

$$A_4 = \begin{pmatrix} 4 & 8 & 12 & -8 \\ 0 & 5 & 10 & -10 \\ 0 & 0 & -6 & 6 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad \Lambda_3 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ -3/4 & 0 & 0 & 0 \\ 1/4 & 0 & 0 & 0 \\ 1/2 & -1/5 & 1/3 & 0 \end{pmatrix}$$

$$P_3 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

Consequently, adding the identity to Λ_3, we obtain

$$L = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -3/4 & 1 & 0 & 0 \\ 1/4 & 0 & 1 & 0 \\ 1/2 & -1/5 & 1/3 & 1 \end{pmatrix}, \qquad U = \begin{pmatrix} 4 & 8 & 12 & -8 \\ 0 & 5 & 10 & -10 \\ 0 & 0 & -6 & 6 \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

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

We check that

$$PA = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 2 & -3 & 4 \\ 4 & 8 & 12 & -8 \\ 2 & 3 & 2 & 1 \\ -3 & -1 & 1 & -4 \end{pmatrix}$$

$$= \begin{pmatrix} 4 & 8 & 12 & -8 \\ -3 & -1 & 1 & -4 \\ 1 & 2 & -3 & 4 \\ 2 & 3 & 2 & 1 \end{pmatrix},$$

and that

$$LU = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -3/4 & 1 & 0 & 0 \\ 1/4 & 0 & 1 & 0 \\ 1/2 & -1/5 & 1/3 & 1 \end{pmatrix} \begin{pmatrix} 4 & 8 & 12 & -8 \\ 0 & 5 & 10 & -10 \\ 0 & 0 & -6 & 6 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 4 & 8 & 12 & -8 \\ -3 & -1 & 1 & -4 \\ 1 & 2 & -3 & 4 \\ 2 & 3 & 2 & 1 \end{pmatrix} = PA.$$

Note that if one willing to *overwrite* the lower triangular part of the evolving matrix A, one can store the evolving Λ there, since these entries will eventually be zero anyway!

There is also no need to save explicitly the permutation matrix P. One could instead record the permutation steps in an extra column (record the vector $(\pi(1), \ldots, \pi(n))$ corresponding to the permutation π applied to the rows).

We let the reader write such a bold and space-efficient version of LU-decomposition!

Proposition 4.6. *If an invertible symmetric matrix A has an LU-decomposition, then A has a factorization of the form*

$$A = LDL^\top,$$

where L is a lower-triangular matrix whose diagonal entries are equal to 1, and where D consists of the pivots. Furthermore, such a decomposition is unique.

Remark: It can be shown that Gaussian elimination + back-substitution requires $n^3/3+O(n^2)$ additions, $n^3/3+O(n^2)$ multiplications and $n^2/2+O(n)$ divisions.

Let us now briefly comment on the choice of a pivot.

Although theoretically, any pivot can be chosen, the possibility of roundoff errors implies that it is *not a good idea to pick very small pivots*. The following example illustrates this point.

$$\begin{aligned} 10^{-4}x + y &= 1 \\ x + y &= 2. \end{aligned}$$

Since 10^{-4} is nonzero, it can be taken as pivot, and we get

$$\begin{aligned} 10^{-4}x + y &= 1 \\ (1 - 10^4)y &= 2 - 10^4. \end{aligned}$$

Thus, the exact solution is

$$x = \frac{10^4}{10^4 - 1}, \quad y = \frac{10^4 - 2}{10^4 - 1}.$$

However, if roundoff takes place on the fourth digit, then $10^4 - 1 = 9999$ and $10^4 - 2 = 9998$ will be rounded off both to 9990, and then, the solution is $x = 0$ and $y = 1$, very far from the exact solution where $x \approx 1$ and $y \approx 1$.

The problem is that we picked a *very small pivot*.

If instead we permute the equations, the pivot is 1, and after elimination, we get the system

$$\begin{aligned} x + \quad y \quad &= \quad 2 \\ (1 - 10^{-4})y &= 1 - 2 \times 10^{-4}. \end{aligned}$$

This time, $1 - 10^{-4} = 0.9999$ and $1 - 2 \times 10^{-4} = 0.9998$ are rounded off to 0.999 and the solution is
$x = 1, y = 1$, much closer to the exact solution.

To remedy this problem, one may use the strategy of *partial pivoting*.

This consists of choosing during step k $(1 \leq k \leq n-1)$ one of the entries $a_{i\,k}^{(k)}$ such that

$$|a_{i\,k}^{(k)}| = \max_{k \leq p \leq n} |a_{p\,k}^{(k)}|.$$

By maximizing the value of the pivot, we avoid dividing by undesirably small pivots.

Remark: A matrix, A, is called *strictly column diagonally dominant* iff

$$|a_{j\,j}| > \sum_{i=1,\, i \neq j}^{n} |a_{i\,j}|, \quad \text{for } j = 1, \ldots, n$$

(resp. *strictly row diagonally dominant* iff

$$|a_{i\,i}| > \sum_{j=1,\, j \neq i}^{n} |a_{i\,j}|, \quad \text{for } i = 1, \ldots, n.)$$

It has been known for a long time (before 1900, say by Hadamard) that if a matrix, A, is strictly column diagonally dominant (resp. strictly row diagonally dominant), then it is invertible. (This is a good exercise, try it!)

It can also be shown that if A is strictly column diagonally dominant, then Gaussian elimination with partial pivoting does not actually require pivoting.

Another strategy, called *complete pivoting*, consists in choosing some entry $a_{i\,j}^{(k)}$, where $k \leq i, j \leq n$, such that

$$|a_{i\,j}^{(k)}| = \max_{k \leq p,q \leq n} |a_{p\,q}^{(k)}|.$$

However, in this method, if the chosen pivot is not in column k, it is also necessary to *permute columns*.

This is achieved by multiplying on the right by a permutation matrix.

However, complete pivoting tends to be too expensive in practice, and partial pivoting is the method of choice.

A special case where the LU-factorization is particularly efficient is the case of tridiagonal matrices, which we now consider.

4.3 Gaussian Elimination of Tridiagonal Matrices

Consider the tridiagonal matrix

$$A = \begin{pmatrix} b_1 & c_1 & & & & \\ a_2 & b_2 & c_2 & & & \\ & a_3 & b_3 & c_3 & & \\ & \ddots & \ddots & \ddots & & \\ & & a_{n-2} & b_{n-2} & c_{n-2} & \\ & & & a_{n-1} & b_{n-1} & c_{n-1} \\ & & & & a_n & b_n \end{pmatrix}.$$

Define the sequence

$$\begin{aligned} \delta_0 &= 1, \\ \delta_1 &= b_1, \\ \delta_k &= b_k\delta_{k-1} - a_k c_{k-1}\delta_{k-2}, \quad 2 \leq k \leq n. \end{aligned}$$

Proposition 4.7. *If A is the tridiagonal matrix above, then $\delta_k = \det(A[1..k, 1..k])$, for $k = 1, \ldots, n$.*

Theorem 4.8. *If A is the tridiagonal matrix above and $\delta_k \neq 0$ for $k = 1, \ldots, n$, then A has the following LU-factorization:*

$$A = \begin{pmatrix} 1 & & & & & \\ a_2\dfrac{\delta_0}{\delta_1} & 1 & & & & \\ & a_3\dfrac{\delta_1}{\delta_2} & 1 & & & \\ & & \ddots & \ddots & & \\ & & & a_{n-1}\dfrac{\delta_{n-3}}{\delta_{n-2}} & 1 & \\ & & & & a_n\dfrac{\delta_{n-2}}{\delta_{n-1}} & 1 \end{pmatrix} \begin{pmatrix} \dfrac{\delta_1}{\delta_0} & c_1 & & & & \\ & \dfrac{\delta_2}{\delta_1} & c_2 & & & \\ & & \dfrac{\delta_3}{\delta_2} & c_3 & & \\ & & & \ddots & \ddots & \\ & & & & \dfrac{\delta_{n-1}}{\delta_{n-2}} & c_{n-1} \\ & & & & & \dfrac{\delta_n}{\delta_{n-1}} \end{pmatrix}.$$

It follows that there is a simple method to solve a linear system, $Ax = d$, where A is tridiagonal (and $\delta_k \neq 0$ for $k = 1, \ldots, n$).

For this, it is convenient to "squeeze" the diagonal matrix, Δ, defined such that $\Delta_{k\,k} = \delta_k/\delta_{k-1}$, into the factorization so that $A = (L\Delta)(\Delta^{-1}U)$, and if we let

$$\begin{aligned} z_1 &= \frac{c_1}{b_1}, \\ z_k &= c_k\,\frac{\delta_{k-1}}{\delta_k}, \quad 2 \leq k \leq n-1, \\ z_n &= \frac{\delta_n}{\delta_{n-1}} = b_n - a_n z_{n-1}, \end{aligned}$$

$A = (L\Delta)(\Delta^{-1}U)$ is written as

$$A = \begin{pmatrix} \frac{c_1}{z_1} & & & & \\ a_2 & \frac{c_2}{z_2} & & & \\ & a_3 & \frac{c_3}{z_3} & & \\ & & \ddots & \ddots & \\ & & & a_{n-1} & \frac{c_{n-1}}{z_{n-1}} \\ & & & & a_n \quad z_n \end{pmatrix}$$

$$\begin{pmatrix} 1 & z_1 & & & & \\ & 1 & z_2 & & & \\ & & 1 & z_3 & & \\ & & & \ddots & \ddots & \\ & & & & 1 & z_{n-2} \\ & & & & & 1 \quad z_{n-1} \\ & & & & & & 1 \end{pmatrix}.$$

As a consequence, the system $Ax = d$ can be solved by constructing three sequences: First, the sequence

$$\begin{aligned} z_1 &= \frac{c_1}{b_1}, \\ z_k &= \frac{c_k}{b_k - a_k z_{k-1}}, \quad k = 2, \ldots, n-1, \\ z_n &= b_n - a_n z_{n-1}, \end{aligned}$$

corresponding to the recurrence $\delta_k = b_k\delta_{k-1} - a_k c_{k-1}\delta_{k-2}$ and obtained by dividing both sides of this equation by δ_{k-1}, next

$$w_1 = \frac{d_1}{b_1}, \quad w_k = \frac{d_k - a_k w_{k-1}}{b_k - a_k z_{k-1}}, \quad k = 2, \ldots, n,$$

corresponding to solving the system $L\Delta w = d$, and finally

$$x_n = w_n, \quad x_k = w_k - z_k x_{k+1}, \quad k = n-1, n-2, \ldots, 1,$$

corresponding to solving the system $\Delta^{-1}Ux = w$.

Remark: It can be verified that this requires $3(n-1)$ additions, $3(n-1)$ multiplications, and $2n$ divisions, a total of $8n-6$ operations, which is much less that the $O(2n^3/3)$ required by Gaussian elimination in general.

We now consider the special case of symmetric positive definite matrices (SPD matrices).

Recall that an $n \times n$ symmetric matrix, A, is *positive definite* iff

$$x^\top Ax > 0 \quad \text{for all } x \in \mathbb{R}^n \text{ with } x \neq 0.$$

Equivalently, A is symmetric positive definite iff all its eigenvalues are strictly positive.

The following facts about a symmetric positive definite matrice, A, are easily established:

(1) The matrix A is invertible. (Indeed, if $Ax = 0$, then $x^\top Ax = 0$, which implies $x = 0$.)

(2) We have $a_{i\,i} > 0$ for $i = 1, \ldots, n$. (Just observe that for $x = e_i$, the ith canonical basis vector of $\mathbb{R}^n$, we have $e_i^\top A e_i = a_{i\,i} > 0$.)

(3) For every $n \times n$ invertible matrix, Z, the matrix $Z^\top AZ$ is symmetric positive definite iff A is symmetric positive definite.

Next, we prove that a symmetric positive definite matrix has a special LU-factorization of the form $A = BB^\top$, where B is a lower-triangular matrix whose diagonal elements are strictly positive.

This is the *Cholesky factorization*.

4.4 SPD Matrices and the Cholesky Decomposition

First, we note that a symmetric positive definite matrix satisfies the condition of Proposition 4.2.

Proposition 4.9. *If A is a symmetric positive definite matrix, then $A[1..k, 1..k]$ is symmetric positive definite, and thus invertible for $k = 1, \ldots, n$.*

Let A be a symmetric positive definite matrix and write

$$A = \begin{pmatrix} a_{11} & W^\top \\ W & C \end{pmatrix},$$

where C is an $(n-1) \times (n-1)$ symmetric matrix and W is an $(n-1) \times 1$ matrix.

Since A is symmetric positive definite, $a_{11} > 0$, and we can compute $\alpha = \sqrt{a_{11}}$. The trick is that we can factor A uniquely as

$$\begin{aligned} A &= \begin{pmatrix} a_{11} & W^\top \\ W & C \end{pmatrix} \\ &= \begin{pmatrix} \alpha & 0 \\ W/\alpha & I \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & C - WW^\top/a_{11} \end{pmatrix} \begin{pmatrix} \alpha & W^\top/\alpha \\ 0 & I \end{pmatrix}, \end{aligned}$$

i.e., as $A = B_1 A_1 B_1^\top$, where B_1 is lower-triangular with positive diagonal entries.

Thus, B_1 is invertible, and by fact (3) above, A_1 is also symmetric positive definite.

Theorem 4.10. *(Cholesky Factorization) Let A be a symmetric positive definite matrix. Then, there is some lower-triangular matrix, B, so that $A = BB^\top$. Furthermore, B can be chosen so that its diagonal elements are strictly positive, in which case, B is unique.*

Remark: If $A = BB^\top$, where B is any invertible matrix, then A is symmetric positive definite.

The proof of Theorem 4.10 immediately yields an algorithm to compute B from A. For $j = 1, \ldots, n$,

$$b_{j\,j} = \left(a_{j\,j} - \sum_{k=1}^{j-1} b_{j\,k}^2\right)^{1/2},$$

and for $i = j+1, \ldots, n$ (and $j = 1, \ldots, n-1$)

$$b_{i\,j} = \left(a_{i\,j} - \sum_{k=1}^{j-1} b_{i\,k} b_{j\,k}\right) / b_{j\,j}.$$

The above formulae are used to compute the jth column of B from top-down, using the first $j-1$ columns of B previously computed, and the matrix A.

For example, if

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

we find that

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

The Cholesky factorization can be used to solve linear systems, $Ax = b$, where A is symmetric positive definite:

Solve the two systems $Bw = b$ and $B^\top x = w$.

Remark: It can be shown that this methods requires $n^3/6 + O(n^2)$ additions, $n^3/6 + O(n^2)$ multiplications, $n^2/2 + O(n)$ divisions, and $O(n)$ square root extractions.

Thus, the Cholesky method requires half of the number of operations required by Gaussian elimination (since Gaussian elimination requires $n^3/3 + O(n^2)$ additions, $n^3/3 + O(n^2)$ multiplications, and $n^2/2 + O(n)$ divisions).

It also requires half of the space (only B is needed, as opposed to both L and U).

Furthermore, it can be shown that Cholesky's method is numerically stable.

We now give three more criteria for a symmetric matrix to be positive definite.

Proposition 4.11. *Let A be any $n \times n$ symmetric matrix. The following conditions are equivalent:*

(a) A is positive definite.

(b) All principal minors of A are positive; that is: $\det(A[1..k, 1..k]) > 0$ for $k = 1, \ldots, n$ (Sylvester's criterion).

(c) A has an LU-factorization and all pivots are positive.

(d) A has an $LDL^\top$-factorization and all pivots in D are positive.

For more on the stability analysis and efficient implementation methods of Gaussian elimination, LU-factoring and Cholesky factoring, see Demmel [11], Trefethen and Bau [34], Ciarlet [9], Golub and Van Loan [17], Strang [31, 32], and Kincaid and Cheney [22].

4.5 Reduced Row Echelon Form

Gaussian elimination described in Section 4.2 can also be applied to rectangular matrices.

This yields a method for determining whether a system $Ax = b$ is solvable, and a description of all the solutions when the system is solvable, for any rectangular $m \times n$ matrix A.

It turns out that the discussion is simpler if we rescale all pivots to be 1, and for this we need a third kind of elementary matrix.

For any $\lambda \neq 0$, let $E_{i,\lambda}$ be the $n \times n$ diagonal matrix

$$E_{i,\lambda} = \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & & \lambda & & & \\ & & & & 1 & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{pmatrix},$$

with $(E_{i,\lambda})_{ii} = \lambda$ $(1 \leq i \leq n)$.

Note that $E_{i,\lambda}$ is also given by

$$E_{i,\lambda} = I + (\lambda - 1)e_{i\,i},$$

and that $E_{i,\lambda}$ is invertible with

$$E_{i,\lambda}^{-1} = E_{i,\lambda^{-1}}.$$

Now, after $k-1$ elimination steps, if the bottom portion

$$(a_{kk}^{(k)}, a_{k+1k}^{(k)}, \ldots, a_{mk}^{(k)})$$

of the kth column of the current matrix A_k is nonzero so that a pivot π_k can be chosen, after a permutation of rows if necessary, we also divide row k by π_k to obtain the pivot 1, and not only do we zero all the entries $i = k+1, \ldots, m$ in column k, but also all the entries $i = 1, \ldots, k-1$, so that the only nonzero entry in column k is a 1 in row k.

These row operations are achieved by multiplication on the left by elementary matrices.

If $a_{kk}^{(k)} = a_{k+1k}^{(k)} = \cdots = a_{mk}^{(k)} = 0$, we move on to column $k+1$.

When the kth column contains a pivot, the kth stage of the procedure for converting a matrix to *rref* consists of the following three steps illustrated below:

$$\begin{pmatrix} 1 & \times & 0 & \times & \times & \times & \times \\ 0 & 0 & 1 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & a_{ik}^{(k)} & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times & \times \end{pmatrix} \overset{\text{pivot}}{\Longrightarrow} \begin{pmatrix} 1 & \times & 0 & \times & \times & \times & \times \\ 0 & 0 & 1 & \times & \times & \times & \times \\ 0 & 0 & 0 & a_{ik}^{(k)} & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times & \times \end{pmatrix}$$

$$\overset{\text{rescale}}{\Longrightarrow}$$

$$\begin{pmatrix} 1 & \times & 0 & \times & \times & \times & \times \\ 0 & 0 & 1 & \times & \times & \times & \times \\ 0 & 0 & 0 & 1 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times & \times \end{pmatrix} \overset{\text{elim}}{\Longrightarrow} \begin{pmatrix} 1 & \times & 0 & \mathbf{0} & \times & \times & \times \\ 0 & 0 & 1 & \mathbf{0} & \times & \times & \times \\ 0 & 0 & 0 & \mathbf{1} & \times & \times & \times \\ 0 & 0 & 0 & \mathbf{0} & \times & \times & \times \\ 0 & 0 & 0 & \mathbf{0} & \times & \times & \times \\ 0 & 0 & 0 & \mathbf{0} & \times & \times & \times \end{pmatrix}.$$

If the kth column does not contain a pivot, we simply move on to the next column.

Here is an example illustrating this process: Starting from the matrix

$$A_1 = \begin{pmatrix} 1 & 0 & 2 & 1 & 5 \\ 1 & 1 & 5 & 2 & 7 \\ 1 & 2 & 8 & 4 & 12 \end{pmatrix}$$

we perform the following steps

$$A_1 \longrightarrow A_2 = \begin{pmatrix} 1 & 0 & 2 & 1 & 5 \\ 0 & 1 & 3 & 1 & 2 \\ 0 & 2 & 6 & 3 & 7 \end{pmatrix},$$

by subtracting row 1 from row 2 and row 3;

$$A_2 \longrightarrow \begin{pmatrix} 1 & 0 & 2 & 1 & 5 \\ 0 & 2 & 6 & 3 & 7 \\ 0 & 1 & 3 & 1 & 2 \end{pmatrix} \longrightarrow \begin{pmatrix} 1 & 0 & 2 & 1 & 5 \\ 0 & 1 & 3 & 3/2 & 7/2 \\ 0 & 1 & 3 & 1 & 2 \end{pmatrix}$$
$$\longrightarrow A_3 = \begin{pmatrix} 1 & 0 & 2 & 1 & 5 \\ 0 & 1 & 3 & 3/2 & 7/2 \\ 0 & 0 & 0 & -1/2 & -3/2 \end{pmatrix},$$

after choosing the pivot 2 and permuting row 2 and row 3, dividing row 2 by 2, and subtracting row 2 from row 3;

$$A_3 \longrightarrow \begin{pmatrix} 1 & 0 & 2 & 1 & 5 \\ 0 & 1 & 3 & 3/2 & 7/2 \\ 0 & 0 & 0 & 1 & 3 \end{pmatrix} \longrightarrow A_4 = \begin{pmatrix} 1 & 0 & 2 & 0 & 2 \\ 0 & 1 & 3 & 0 & -1 \\ 0 & 0 & 0 & 1 & 3 \end{pmatrix},$$

after dividing row 3 by $-1/2$, subtracting row 3 from row 1, and subtracting $(3/2) \times$ row 3 from row 2.

It is clear that columns $1, 2$ and 4 are linearly independent, that column 3 is a linear combination of columns 1 and 2, and that column 5 is a linear combinations of columns $1, 2, 4$.

The result is that after performing such elimination steps, we obtain a matrix that has a special shape known as a *reduced row echelon matrix*, for short *rref*.

In general, the sequence of steps leading to a reduced echelon matrix is not unique.

For example, we could have chosen 1 instead of 2 as the second pivot in matrix A_2.

Nevertheless, the reduced row echelon matrix obtained from any given matrix is unique; that is, it does not depend on the the sequence of steps that are followed during the reduction process.

If we want to solve a linear system of equations of the form $Ax = b$, we apply elementary row operations to both the matrix A and the right-hand side b.

To do this conveniently, we form the *augmented matrix* (A, b), which is the $m \times (n+1)$ matrix obtained by adding b as an extra column to the matrix A.

For example if

$$A = \begin{pmatrix} 1 & 0 & 2 & 1 \\ 1 & 1 & 5 & 2 \\ 1 & 2 & 8 & 4 \end{pmatrix} \quad \text{and} \quad b = \begin{pmatrix} 5 \\ 7 \\ 12 \end{pmatrix},$$

then the augmented matrix is

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

Now, for any matrix M, since

$$M(A, b) = (MA, Mb),$$

performing elementary row operations on (A, b) is equivalent to simultaneously performing operations on both A and b.

For example, consider the system

$$\begin{array}{rcrcrcl} x_1 & & & + & 2x_3 & + & x_4 & = & 5 \\ x_1 & + & x_2 & + & 5x_3 & + & 2x_4 & = & 7 \\ x_1 & + & 2x_2 & + & 8x_3 & + & 4x_4 & = & 12. \end{array}$$

Its augmented matrix is the matrix

$$(A, b) = \begin{pmatrix} 1 & 0 & 2 & 1 & 5 \\ 1 & 1 & 5 & 2 & 7 \\ 1 & 2 & 8 & 4 & 12 \end{pmatrix}$$

considered above, so the reduction steps applied to this matrix yield the system

$$\begin{array}{rcrcrcl} x_1 & & & + & 2x_3 & & = & 2 \\ & & x_2 & + & 3x_3 & & = & -1 \\ & & & & & x_4 & = & 3. \end{array}$$

This reduced system has the same set of solutions as the original, and obviously x_3 can be chosen arbitrarily. Therefore, our system has infinitely many solutions given by

$$x_1 = 2 - 2x_3, \quad x_2 = -1 - 3x_3, \quad x_4 = 3,$$

where x_3 is arbitrary.

The following proposition shows that the set of solutions of a system $Ax = b$ is preserved by any sequence of row operations.

Proposition 4.12. *Given any $m \times n$ matrix A and any vector $b \in \mathbb{R}^m$, for any sequence of elementary row operations $E_1, \ldots, E_k$, if $P = E_k \cdots E_1$ and $(A', b') = P(A, b)$, then the solutions of $Ax = b$ are the same as the solutions of $A'x = b'$.*

Another important fact is this:

Proposition 4.13. *Given a $m \times n$ matrix A, for any sequence of row operations $E_1, \ldots, E_k$, if $P = E_k \cdots E_1$ and $B = PA$, then the subspaces spanned by the rows of A and the rows of B are identical. Therefore, A and B have the same row rank. Furthermore, the matrices A and B also have the same (column) rank.*

Remark: The subspaces spanned by the columns of A and B can be different! However, their dimension must be the same.

We already know from Proposition 3.24 that the row rank is equal to the column rank.

We will see that the reduction to row echelon form provides another proof of this important fact.

Definition 4.1. A $m \times n$ matrix A is a *reduced row echelon matrix* iff the following conditions hold:

(a) The first nonzero entry in every row is 1. This entry is called a *pivot*.

(b) The first nonzero entry of row $i+1$ is to the right of the first nonzero entry of row i.

(c) The entries above a pivot are zero.

If a matrix satisfies the above conditions, we also say that it is in *reduced row echelon form*, for short *rref*.

Note that condition (b) implies that the entries below a pivot are also zero. For example, the matrix

$$A = \begin{pmatrix} 1 & 6 & 0 & 1 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

is a reduced row echelon matrix.

In general, a matrix in *rref* has the following shape:

$$\begin{pmatrix} 1 & 0 & 0 & \times & \times & 0 & 0 & \times \\ 0 & 1 & 0 & \times & \times & 0 & 0 & \times \\ 0 & 0 & 1 & \times & \times & 0 & 0 & \times \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & \times \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & \times \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

if the last row consists of zeros, or

$$\begin{pmatrix} 1 & 0 & 0 & \times & \times & 0 & 0 & \times & 0 & \times \\ 0 & 1 & 0 & \times & \times & 0 & 0 & \times & 0 & \times \\ 0 & 0 & 1 & \times & \times & 0 & 0 & \times & 0 & \times \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & \times & 0 & \times \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & \times & \times & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \times \end{pmatrix}$$

if the last row contains a pivot.

Proposition 4.14. *Given any $m \times n$ matrix A, there is a sequence of row operations $E_1, \ldots, E_k$ such that if $P = E_k \cdots E_1$, then $U = PA$ is a reduced row echelon matrix.*

Remark: There is a `Matlab` function named `rref` that converts any matrix to its reduced row echelon form.

If A is any matrix and if R is a reduced row echelon form of A, the second part of Proposition 4.13 can be sharpened a little.

Namely, *the rank of A is equal to the number of pivots in R*.

Given a system of the form $Ax = b$, we can apply the reduction procedure to the augmented matrix (A, b) to obtain a reduced row echelon matrix (A', b') such that the system $A'x = b'$ has the same solutions as the original system $Ax = b$.

The advantage of the reduced system $A'x = b'$ is that there is a simple test to check whether this system is solvable, and to find its solutions if it is solvable.

Indeed, if any row of the matrix A' is zero and if the corresponding entry in b' is nonzero, then it is a pivot and we have the "equation"

$$0 = 1,$$

which means that the system $A'x = b'$ has no solution.

On the other hand, if there is no pivot in b', then for every row i in which $b'_i \neq 0$, there is some column j in A' where the entry on row i is 1 (a pivot).

Consequently, we can assign arbitrary values to the variable x_k if column k does not contain a pivot, and then solve for the pivot variables.

For example, if we consider the reduced row echelon matrix

$$(A', b') = \begin{pmatrix} 1 & 6 & 0 & 1 & 0 \\ 0 & 0 & 1 & 2 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix},$$

there is no solution to $A'x = b'$ because the third equation is $0 = 1$.

On the other hand, the reduced system

$$(A', b') = \begin{pmatrix} 1 & 6 & 0 & 1 & 1 \\ 0 & 0 & 1 & 2 & 3 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

has solutions. We can pick the variables x_2, x_4 corresponding to nonpivot columns arbitrarily, and then solve for x_3 (using the second equation) and x_1 (using the first equation).

The above reasoning proved the following theorem:

Theorem 4.15. *Given any system $Ax = b$ where A is a $m \times n$ matrix, if the augmented matrix (A, b) is a reduced row echelon matrix, then the system $Ax = b$ has a solution iff there is no pivot in b. In that case, an arbitrary value can be assigned to the variable x_j if column j does not contain a pivot.*

Nonpivot variables are often called *free variables*.

Putting Proposition 4.14 and Theorem 4.15 together we obtain a criterion to decide whether a system $Ax = b$ has a solution:

Convert the augmented system (A, b) to a row reduced echelon matrix (A', b') and check whether b' has no pivot.

If we have a *homogeneous system* $Ax = 0$, which means that $b = 0$, of course $x = 0$ is always a solution, but Theorem 4.15 implies that if the system $Ax = 0$ has more variables than equations, then it has some nonzero solution (we call it a *nontrivial solution*).

Proposition 4.16. *Given any homogeneous system $Ax = 0$ of m equations in n variables, if $m < n$, then there is a nonzero vector $x \in \mathbb{R}^n$ such that $Ax = 0$.*

Theorem 4.15 can also be used to characterize when a square matrix is invertible. First, note the following simple but important fact:

If a square $n \times n$ matrix A is a row reduced echelon matrix, then either A is the identity or the bottom row of A is zero.

Proposition 4.17. *Let A be a square matrix of dimension n. The following conditions are equivalent:*

(a) The matrix A can be reduced to the identity by a sequence of elementary row operations.

(b) The matrix A is a product of elementary matrices.

(c) The matrix A is invertible.

(d) The system of homogeneous equations $Ax = 0$ has only the trivial solution $x = 0$.

Proposition 4.17 yields a method for computing the inverse of an invertible matrix A: reduce A to the identity using elementary row operations, obtaining

$$E_p \cdots E_1 A = I.$$

Multiplying both sides by A^{-1} we get

$$A^{-1} = E_p \cdots E_1.$$

From a practical point of view, we can build up the product $E_p \cdots E_1$ by reducing to row echelon form the augmented $n \times 2n$ matrix (A, I_n) obtained by adding the n columns of the identity matrix to A.

This is just another way of performing the Gauss–Jordan procedure.

Here is an example: let us find the inverse of the matrix

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

We form the 2×4 block matrix

$$(A, I) = \begin{pmatrix} 5 & 4 & 1 & 0 \\ 6 & 5 & 0 & 1 \end{pmatrix}$$

and apply elementary row operations to reduce A to the identity.

For example:

$$(A, I) = \begin{pmatrix} 5 & 4 & 1 & 0 \\ 6 & 5 & 0 & 1 \end{pmatrix} \longrightarrow \begin{pmatrix} 5 & 4 & 1 & 0 \\ 1 & 1 & -1 & 1 \end{pmatrix}$$

by subtracting row 1 from row 2,

$$\begin{pmatrix} 5 & 4 & 1 & 0 \\ 1 & 1 & -1 & 1 \end{pmatrix} \longrightarrow \begin{pmatrix} 1 & 0 & 5 & -4 \\ 1 & 1 & -1 & 1 \end{pmatrix}$$

by subtracting $4 \times$ row 2 from row 1,

$$\begin{pmatrix} 1 & 0 & 5 & -4 \\ 1 & 1 & -1 & 1 \end{pmatrix} \longrightarrow \begin{pmatrix} 1 & 0 & 5 & -4 \\ 0 & 1 & -6 & 5 \end{pmatrix} = (I, A^{-1}),$$

by subtracting row 1 from row 2. Thus

$$A^{-1} = \begin{pmatrix} 5 & -4 \\ -6 & 5 \end{pmatrix}.$$

Proposition 4.17 can also be used to give an elementary proof of the fact that if a square matrix A has a left inverse B (resp. a right inverse B), so that $BA = I$ (resp. $AB = I$), then A is invertible and $A^{-1} = B$. This is an interesting exercise, try it!

For the sake of completeness, we prove that the reduced row echelon form of a matrix is unique.

Proposition 4.18. *Let A be any $m \times n$ matrix. If U and V are two reduced row echelon matrices obtained from A by applying two sequences of elementary row operations $E_1, \ldots, E_p$ and $F_1, \ldots, F_q$, so that*

$$U = E_p \cdots E_1 A \quad \textit{and} \quad V = F_q \cdots F_1 A,$$

then $U = V$ and $E_p \cdots E_1 = F_q \cdots F_1$. In other words, the reduced row echelon form of any matrix is unique.

The reduction to row echelon form also provides a method to describe the set of solutions of a linear system of the form $Ax = b$.

Proposition 4.19. *Let A be any $m \times n$ matrix and let $b \in \mathbb{R}^m$ be any vector. If the system $Ax = b$ has a solution, then the set Z of all solutions of this system is the set*

$$Z = x_0 + \mathrm{Ker}\,(A) = \{x_0 + x \mid Ax = 0\},$$

where $x_0 \in \mathbb{R}^n$ is any solution of the system $Ax = b$, which means that $Ax_0 = b$ (x_0 is called a special solution), and where $\mathrm{Ker}\,(A) = \{x \in \mathbb{R}^n \mid Ax = 0\}$, the set of solutions of the homogeneous system associated with $Ax = b$.

Given a linear system $Ax = b$, reduce the augmented matrix (A, b) to its row echelon form (A', b').

As we showed before, the system $Ax = b$ has a solution iff b' contains no pivot. Assume that this is the case.

Then, if (A', b') has r pivots, which means that A' has r pivots since b' has no pivot, we know that the first r columns of I_m appear in A'.

We can permute the columns of A' and renumber the variables in x correspondingly so that the first r columns of I_m match the first r columns of A', and then our reduced echelon matrix is of the form (R, b') with

$$R = \begin{pmatrix} I_r & F \\ 0_{m-r,r} & 0_{m-r,n-r} \end{pmatrix}$$

and

$$b' = \begin{pmatrix} d \\ 0_{m-r} \end{pmatrix},$$

where F is a $r \times (n-r)$ matrix and $d \in \mathbb{R}^r$. Note that R has $m - r$ zero rows.

Then, because

$$\begin{pmatrix} I_r & F \\ 0_{m-r,r} & 0_{m-r,n-r} \end{pmatrix} \begin{pmatrix} d \\ 0_{n-r} \end{pmatrix} = \begin{pmatrix} d \\ 0_{m-r} \end{pmatrix} = b',$$

we see that

$$x_0 = \begin{pmatrix} d \\ 0_{n-r} \end{pmatrix}$$

is a special solution of $Rx = b'$, and thus to $Ax = b$.

In other words, we get a special solution by assigning the first r components of b' to the pivot variables and setting the nonpivot variables (the *free variables*) to zero.

We can also find a basis of the kernel (nullspace) of A using F.

If $x = (u, v)$ is in the kernel of A, with $u \in \mathbb{R}^r$ and $v \in \mathbb{R}^{n-r}$, then x is also in the kernel of R, which means that $Rx = 0$; that is,

$$\begin{pmatrix} I_r & F \\ 0_{m-r,r} & 0_{m-r,n-r} \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} u + Fv \\ 0_{m-r} \end{pmatrix} = \begin{pmatrix} 0_r \\ 0_{m-r} \end{pmatrix}.$$

Therefore, $u = -Fv$, and $\mathrm{Ker}\,(A)$ consists of all vectors of the form

$$\begin{pmatrix} -Fv \\ v \end{pmatrix} = \begin{pmatrix} -F \\ I_{n-r} \end{pmatrix} v,$$

for any arbitrary $v \in \mathbb{R}^{n-r}$.

It follows that the $n - r$ columns of the matrix

$$N = \begin{pmatrix} -F \\ I_{n-r} \end{pmatrix}$$

form a basis of the kernel of A.

In summary, if $N^1, \ldots, N^{n-r}$ are the columns of N, then the general solution of the equation $Ax = b$ is given by

$$x = \begin{pmatrix} d \\ 0_{n-r} \end{pmatrix} + x_{r+1}N^1 + \cdots + x_n N^{n-r},$$

where $x_{r+1}, \ldots, x_n$ are the free variables, that is, the non-pivot variables.

In the general case where the columns corresponding to pivots are mixed with the columns corresponding to free variables, we find the special solution as follows.

Let $i_1 < \cdots < i_r$ be the indices of the columns corresponding to pivots. Then, assign b'_k to the pivot variable x_{i_k} for $k = 1, \ldots, r$, and set all other variables to 0.

To find a basis of the kernel, we form the $n - r$ vectors N^k obtained as follows.

Let $j_1 < \cdots < j_{n-r}$ be the indices of the columns corresponding to free variables.

For every column j_k corresponding to a free variable ($1 \leq k \leq n - r$), form the vector N^k defined so that the entries $N^k_{i_1}, \ldots, N^k_{i_r}$ are equal to the negatives of the first r entries in column j_k (flip the sign of these entries); let $N^k_{j_k} = 1$, and set all other entries to zero.

The presence of the 1 in position j_k guarantees that $N^1, \ldots,$ N^{n-r} are linearly independent.

An illustration of the above method, consider the problem of finding a basis of the subspace V of $n \times n$ matrices $A \in \mathrm{M}_n(\mathbb{R})$ satisfying the following properties:

1. The sum of the entries in every row has the same value (say c_1);
2. The sum of the entries in every column has the same value (say c_2).

It turns out that $c_1 = c_2$ and that the $2n - 2$ equations corresponding to the above conditions are linearly independent.

By the duality theorem, the dimension of the space V of matrices satisying the above equations is $n^2 - (2n - 2)$.

Let us consider the case $n = 4$. There are 6 equations, and the space V has dimension 10. The equations are

$$\begin{aligned}
a_{11} + a_{12} + a_{13} + a_{14} - a_{21} - a_{22} - a_{23} - a_{24} &= 0\\
a_{21} + a_{22} + a_{23} + a_{24} - a_{31} - a_{32} - a_{33} - a_{34} &= 0\\
a_{31} + a_{32} + a_{33} + a_{34} - a_{41} - a_{42} - a_{43} - a_{44} &= 0\\
a_{11} + a_{21} + a_{31} + a_{41} - a_{12} - a_{22} - a_{32} - a_{42} &= 0\\
a_{12} + a_{22} + a_{32} + a_{42} - a_{13} - a_{23} - a_{33} - a_{43} &= 0\\
a_{13} + a_{23} + a_{33} + a_{43} - a_{14} - a_{24} - a_{34} - a_{44} &= 0.
\end{aligned}$$

Performing rref on the above matrix and applying the method for finding a basis of its kernel, we obtain 10 matrices listed in the notes (linalg.pdf).

Recall that a *magic square* is a square matrix that satisfies the two conditions about the sum of the entries in each row and in each column to be the same number, and also the additional two constraints that the main descending and the main ascending diagonals add up to this common number.

Furthermore, the entries are also required to be positive integers.

For $n = 4$, the additional two equations are

$$\begin{aligned} a_{22} + a_{33} + a_{44} - a_{12} - a_{13} - a_{14} &= 0 \\ a_{41} + a_{32} + a_{23} - a_{11} - a_{12} - a_{13} &= 0, \end{aligned}$$

and the 8 equations stating that a matrix is a magic square are linearly independent.

Again, by running row elimination, we get a basis of the "generalized magic squares" whose entries are not restricted to be positive integers. We find a basis of 8 matrices.

For $n = 3$, we find a basis of 3 matrices.

A magic square is said to be *normal* if its entries are precisely the integers $1, 2 \ldots, n^2$.

Then, since the sum of these entries is

$$1 + 2 + 3 + \cdots + n^2 = \frac{n^2(n^2+1)}{2},$$

and since each row (and column) sums to the same number, this common value (the *magic sum*) is

$$\frac{n(n^2+1)}{2}.$$

It is easy to see that there are no normal magic squares for $n = 2$.

For $n = 3$, the magic sum is 15, for $n = 4$, it is 34, and for $n = 5$, it is 65.

In the case $n = 3$, we have the additional condition that the rows and columns add up to 15, so we end up with a solution parametrized by two numbers x_1, x_2; namely,

$$\begin{pmatrix} x_1 + x_2 - 5 & 10 - x_2 & 10 - x_1 \\ 20 - 2x_1 - x_2 & 5 & 2x_1 + x_2 - 10 \\ x_1 & x_2 & 15 - x_1 - x_2 \end{pmatrix}.$$

Thus, in order to find a normal magic square, we have the additional inequality constraints

$$\begin{aligned} x_1 + x_2 &> 5 \\ x_1 &< 10 \\ x_2 &< 10 \\ 2x_1 + x_2 &< 20 \\ 2x_1 + x_2 &> 10 \\ x_1 &> 0 \\ x_2 &> 0 \\ x_1 + x_2 &< 15, \end{aligned}$$

and all 9 entries in the matrix must be distinct.

After a tedious case analysis, we discover the remarkable fact that there is a unique normal magic square (up to rotations and reflections):

$$\begin{pmatrix} 2 & 7 & 6 \\ 9 & 5 & 1 \\ 4 & 3 & 8 \end{pmatrix}.$$

It turns out that there are 880 different normal magic squares for $n = 4$, and $275,305,224$ normal magic squares for $n = 5$ (up to rotations and reflections).

Finding the number of magic squares for $n > 5$ is an open problem!

Even for $n = 4$, it takes a fair amount of work to enumerate them all!

Instead of performing elementary row operations on a matrix A, we can perform elementary columns operations, which means that we multiply A by elementary matrices on the right.

We can define the notion of a reduced column echelon matrix and show that every matrix can be reduced to a unique reduced column echelon form.

Now, given any $m \times n$ matrix A, if we first convert A to its reduced row echelon form R, it is easy to see that we can apply elementary column operations that will reduce R to a matrix of the form

$$\begin{pmatrix} I_r & 0_{r,n-r} \\ 0_{m-r,r} & 0_{m-r,n-r} \end{pmatrix},$$

where r is the number of pivots (obtained during the row reduction).

Therefore, for every $m \times n$ matrix A, there exist two sequences of elementary matrices $E_1, \ldots, E_p$ and $F_1, \ldots, F_q$, such that

$$E_p \cdots E_1 A F_1 \cdots F_q = \begin{pmatrix} I_r & 0_{r,n-r} \\ 0_{m-r,r} & 0_{m-r,n-r} \end{pmatrix}.$$

The matrix on the right-hand side is called the *rank normal form* of A.

Clearly, r is the rank of A. It is easy to see that the rank normal form also yields a proof of the fact that A and its transpose $A^\top$ have the same rank.