Chapter 2

The Solution of Linear Systems $AX = B$

2.1 Upper-triangular Linear Systems

We will now develop the back-substitution algorithm, which is useful for solving a linear system of equations that has an upper-triangular coefficient matrix. This algorithm will be incorporated in the algorithm for solving a general linear system in Section 2.4.

Definition 2.2. An $N \times N$ matrix $A = [a_{ij}]$ is called ***upper triangular*** provided that the elements satisfy $a_{ij} = 0$ whenever $i > j$. The $N \times N$ matrix $A = [a_{ij}]$ is called lower triangular provided that $a_{ij} = 0$ whenever $i < j$.

We will develop a method for constructing the solution to upper-triangular linear systems of equations and leave the investigation of lower-triangular systems to the reader. If A is an upper-triangular matrix, then $AX = B$ is said to an ***upper-***

triangular system of linear equations and has the form

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1N-1}x_{N-1} + a_{1N}x_N &= b_1 \\ a_{22}x_2 + a_{23}x_3 + \cdots + a_{2N-1}x_{N-1} + a_{2N}x_N &= b_2 \\ a_{33}x_3 + \cdots + a_{3N-1}x_{N-1} + a_{3N}x_N &= b_3 \\ \vdots \quad & \quad \vdots \\ a_{N-1N-1}x_{N-1} + a_{N-1N}x_N &= b_{N-1} \\ a_{NN}x_N &= b_N. \end{aligned} \tag{2.1}$$

Theorem 2.5 (Back Substitution). Suppose that $AX = B$ is an upper-triangular system with the form given in (2.1). If

$$a_{kk} \neq 0 \quad \text{for } k = 1, 2, \cdots, N, \tag{2.2}$$

then there exists a unique solution to (2.1).

Constructive Proof. The solution is easy to find. The last equation involves only x_N, so we solve it first:

$$x_N = \frac{b_N}{a_{NN}}. \tag{2.3}$$

Now x_N is known and it can be used in the next-to-last equation:

$$x_{N-1} = \frac{b_{N-1} - a_{N-1N}x_N}{a_{N-1N-1}}. \tag{2.4}$$

Now x_N and x_{N-1} are used to find x_{N-2}:

$$x_{N-2} = \frac{b_{N-2} - a_{N-2N-1}x_{N-1} - a_{N-2N}x_N}{a_{N-2N-2}}. \tag{2.5}$$

Once the value $x_N, x_{N-1}, \ldots, x_{k+1}$ are known, the general step is

$$x_k = \frac{b_k - \sum_{j=k+1}^{N} a_{kj}x_j}{a_{kk}} \quad \text{for } k = N-1, N-2, \ldots, 1. \tag{2.6}$$

The uniqueness of the solution is easy to see. The Nth equation implies that b_N/a_{NN} is the only possible value of x_N. Then finite induction is used to establish that $x_{N-1}, x_{N-2}, \ldots, x_1$ are unique.

Example 2.12. Use back substitution to solve the linear system

$$\begin{aligned} 4x_1 - x_2 + 2x_3 + 3x_4 &= 20 \\ 2x_2 + 7x_3 - 4x_4 &= -7 \\ 6x_3 + 5x_4 &= -4 \\ 3x_4 &= -6. \end{aligned}$$

Solving for x_4 in the last equation yields

$$x_4 = \frac{6}{3} = 2.$$

Using x_2 in the third equation, we obtain

$$x_3 = \frac{6 - 5(2)}{6} = -1.$$

Now $x_3 = -1$ and $x_4 = 2$ are used to find x_2 in the second equation:

$$x_2 = \frac{7 - 7(-1) + 4(2)}{2} = -4.$$

Finally, x_1 is obtained using the first equation:

$$x_1 = \frac{20 + 1(-4) - 2(-1) - 3(2)}{4} = 3.$$

The condition that $a_{kk} \neq 0$ is essential because equation (2.6) involves division by a_{kk}. If this requirement is not fulfilled, either no solution exists or infinitely many solutions exist.

Example 2.13. Show that there is no solution to the linear system

$$\begin{aligned} 4x_1 - x_2 + 2x_3 + 3x_4 &= 20 \\ 0x_2 + 7x_3 - 4x_4 &= -7 \\ 6x_3 + 5x_4 &= -4 \\ 3x_4 &= -6. \end{aligned} \tag{2.7}$$

Using the last equation in (2.7), we must have $x_4 = 2$, which is substituted into the second and third equations to obtain

$$\begin{aligned} 7x_3 - 8 &= -7 \\ 6x_3 + 10 &= -4. \end{aligned} \tag{2.8}$$

The first equation in (2.8) implies that $x_3 = 1/7$, and the second equation implies that $x_3 = -1$. This contradiction leads to the conclusion that there is no solution to the linear system (2.7).

Example 2.14. Show that there are infinitely many solutions to

$$\begin{aligned} 4x_1 - x_2 + 2x_3 + 3x_4 &= 20 \\ 0x_2 + 7x_3 - 0x_4 &= -7 \\ 6x_3 + 5x_4 &= -4 \\ 3x_4 &= -6. \end{aligned} \tag{2.9}$$

Using the last equation in (2.9), we must have $x_4 = 2$, which is substituted into the second and third equations to get $x_3 = -1$, which checks out in both equations. But only two values x_3 and x_4 have been obtained from the second through fourth equations, and when they are substituted into the first equation of (2.9), the result is

$$x_2 = 4x_1 - 16, \tag{2.10}$$

which has infinitely many solutions: hence (2.9) has infinitely many solutions. If we choose a value of x_1 in (2.10), then the value of x_2 is uniquely determined. For example, if we include the equation $x_1 = 2$ in the system (2.9), then from (2.10) we compute $x_2 = -8$.

Theorem 2.4 states that the linear system $AX = B$, where A is an $N \times N$ matrix, has a unique solution if and only if $\det(A) \neq 0$. The following theorem states that if any entry on the main diagonal of an upper-or lower-triangular matrix is zero then $\det(A) = 0$. Thus, by inspecting the coefficient matrices in the previous three examples, it is clear that the system in Example 3.12 has a unique solution, and the systems in Examples 2.13 and 2.14 do not have unique solutions. The proof of Theorem 2.6 can be found in most introductory linear algebra textbooks.

Theorem 2.6. If the $N \times N$ matrix $A = [a_i j]$ is either upper or lower triangular, then

$$\det(A) = a_{11}a_{22}\cdots a_{nn} = \prod_{i=1}^{N} a_{ii}. \tag{2.11}$$

The value of the determinant for the coefficient matrix in Example 2.12 is $\det(A) = 4(-2)(6)(3) = -144$. The value of the determinants of the coefficient matrices in Example 2.13 and 2.14 are both $4(0)(6)(3) = 0$.

The following program will solve the upper-triangular system (1) by the method of back substitution, provided $a_{kk} \neq 0$ for $k = 1, 2, \ldots, N$.

Program 2.1 (Back Substitution). To solve the upper-triangular system $AX = B$ by the method of back substitution. Proceed with the method only if all the diagonal elements are nonzero. First compute $x_N = b_N/a_{NN}$ and the use the rule

$$x_k = \frac{b_k - \sum_{j=k+1} a_{kj}x_j}{a_{kk}} \quad k = N-1, N-2, \ldots, 1.$$

```
function X=backsub(A,B)
%Input   – A is an n × n upper-triangular nonsingular matrix
%        – B is an n × 1 matrix
%Output – X is the solution to the linear system AX=B
%Find the dimension of B and initialize X
n=length(B);
X=zeros(n,1);
```

```
X(n)=B(n)/A(n,n);
for k=n-1:-1:1
    X(k)=(B(k)-A(k,k+1:n)*X(k+1:n))/A(k,k);
end
```

2.1.1 Exercises for Upper-Triangular Linear Systems

in Exercises 1 through 3, solve the upper-triangular system and find the value of the determinant of the coefficient matrix.

2.2 Gaussian Elimination and Pivoting

In this section we develop a scheme for solving a general system $AX = B$ of N equations and N unknowns. The goal is to construct an equivalent upper-triangular system $UX = Y$ that can be solved by the method of Section 2.3.

Two linear systems of dimension $N \times N$ are said to be equivalent provided that their solution sets are the same. Theorems from linear algebra show that when certain transformations are applied to a given system the solution sets do not change.

Theorem 2.7. (Elementary Transformations). The following operations applied to a linear system yield an equivalent system:

Interchange:	The order of two equations can be changed.	(2.12)
Scaling:	Multiplying an equation by a nonzero constant.	(2.13)
Replacement:	An equation can be replaced by the sum of itself and a nonzero multiple of any other equation.	(2.14)

It is common to use (2.14) by replacing an equation with the difference of that equation and a multiple of another equation. There concepts are illustrated in the next example.

Example 2.15. Find the parabola $y = A + Bx + Cx^2$ that passes through the three points $(1, 1), (2, -1)$, and $(3, 1)$.

For each point we obtain an equation relating the value of x to the value of y. The result is the linear system

$$\begin{aligned} A + B + C &= 1 && \text{at } (1,1) \\ A + 2B + 4C &= -1 && \text{at } (2,-1) \\ A + 3B + 9C &= 1 && \text{at } (3,1). \end{aligned} \tag{2.15}$$

The variable A is eliminated from the second and third equations by subtracting the first equation from them. This is an application of replacement transformation (3), and the resulting equivalent linear system is

$$\begin{aligned} A + B + C &= 1 \\ B + 3C &= 2 \\ 2B + 8C &= 0. \end{aligned}$$

The variable B is eliminated from the third equation in (5) by subtracting from it two times the second equation. We arrive at the equivalent upper-triangular system:

$$\begin{aligned} A + B + C &= 1 \\ B + 3C &= 2 \\ 2C &= 4. \end{aligned}$$

The back-substitution algorithm is now used to find the coefficients $C = 4/2 = 1, B = -2 - 3(2) = -8$, and $A = 1 - (-8) - 2 = 7$, and equation of the parabola is $y = 7 - 8x + 2x^2$.

2.3 Triangular Factorization

In Section 3.3 we saw how easy it is to solve an upper-triangular system. Now we introduce the concept of factorization of given matrix A into the product of a lower-triangular matrix L that has 1's along the main diagonal and an upper-triangular matrix U with nonzero diagonal elements. For ease of notation we illustrate the concepts with matrices of dimension 4×4, but they apply to an arbitrary system of dimension $N \times N$.

Definition 3.4. The nonsingular matrix A has a triangular factorization if it can be expressed as the product of a low-triangular matrix L and an upper-triangular matrix U:

$$A = LU.$$

In matrix form, this is written as

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{14} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ m_{21} & 1 & 0 & 0 \\ m_{31} & m_{32} & 1 & 0 \\ m_{41} & m_{42} & m_{43} & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ 0 & u_{22} & u_{23} & u_{14} \\ 0 & 0 & u_{33} & u_{34} \\ 0 & 0 & 0 & u_{44} \end{bmatrix}$$

The condition that A is nonsingular implies that $u_{kk} \neq 0$ for all k. The notation for the entries in L is m_{ij}, and the reason for the choice of m_{ij} instead of l_{ij} will be pointed out soon.

2.3.1 Solution of a Linear System

Suppose that the coefficient matrix A for the linear system $AX = B$ has a triangular factorization (1), then the solution to

$$LUX = B$$

can be obtained by defining $Y = UX$ and then solving two systems:

$$\text{first solve } LY = B \text{ for } Y: \quad \text{then solve } UX = Y \text{ for } X.$$

In equation form, we must first solve the lower triangular system

2.3.2 Triangular Factorization

We now discuss how to obtain the triangular factorization. If row interchanges are not necessary when using Gaussian elimination, the multipliers m_{ij} are the subdiagonal entries in L.

Example 3.21 Use Gaussian elimination to constant the triangular factorization of the matrix

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

Theorem 3.10 (Direct Factorization A=LU. No Row Interchanges). Suppose that Gaussian elimination, without row interchanges, can be successfully performed to solve the general linear system $AX = B$. Then the matrix A can be factored as the product of a lower-triangular matrix L and an upper-triangular matrix U:

$$A = LU.$$

Furthermore L can be constructed to have 1's on its diagonal and U will have nonzero diagonal elements. After finding L and U the solution X is computed in two steps:

1. Solve $LU = B$ for Y using forward substitution.
2. Solve $UX = Y$ for X using back substitution.

2.3.3 Computational Complexity

2.3.4 Permutation matrices

2.3.5 Extending the Gaussian Elimination Process

2.4 Iterative Methods for Linear Systems

The goal of this chapter is extend some of the iterative methods introduced in Chapter 2 to higher dimensions. We consider an extension of fixed-point iteration that applies to systems of linear equations.

2.4.1 Jacobi Iteration

Example 3.26. Consider the system of equations

$$\begin{aligned}
4x - y + z &= -7\\
4x - 8y + z &= 21\\
-2x + y + 5z &= 15
\end{aligned}$$

These equations can be written in the form

$$x = \frac{7 + y - z}{4}$$

$$y = \frac{21 + 4x + z}{8}$$

$$z = \frac{15 + 2x - y}{5}.$$

Table 3.2 Convergence Jacobi iteration for the Linear System (1)

k	x_k	y_k	z_k
0	1.0	2.0	2.0
1	1.75	3.375	3.0
2	1.84375	3.875	3.025
3	1.9625	3.925	2.9625
4	1.99062500	3.97656250	3.00000000
5	1.99414063	3.99531250	3.00093750
⋮	⋮	⋮	⋮
15	1.99999993	3.99999985	2.99999993
⋮	⋮	⋮	⋮
19	2.00000000	4.00000000	3.00000000

This suggests the following Jacobi iterative process:

$$x_{k+1} = \frac{7 + y_k - z_k}{4}$$
$$y_{k+1} = \frac{21 + 4x_k + z_k}{8}$$
$$z_{k+1} = \frac{15 + 2x_k - y_k}{5}.$$

Let us show that if we start with $P_0 = (x_0, y_0, z_0) = (1, 2, 3)$, then the iteration in (3) appears to converge to the solution $(2, 4, 3)$.

Substitute $x_0 = 1, y_0 = 2$, and $z_0 = 2$ into the right-hand side of each equation in (3) to obtain the new values

$$x_1 = \frac{7 + 2 - 2}{4} = 1.75$$

$$y_1 = \frac{21 + 4 + 2}{8} = 3.375$$
$$z_1 = \frac{15 + 2 - 2}{5} = 3.00.$$

The new point $P_1 = (1.75, 3.375, 3.00)$ is closer to $(2, 4, 3)$ then P_0. iteration using (3) generates a sequence of points $\{P_k\}$ that converges to the solution $(2, 4, 3)$ (see Table 3.2).

This process is called Jacobi iteration and can be used to solve certain types of linear systems. After 19 steps, the iteration has converged to the nine-digit machine approximation $(2.00000000, 4.00000000, 3.00000000)$.

Linear systems with as many as $100,000$ variables often arise in the solution of partial differential equations. The coefficient matrices for these systems are sparse; that is, a large percentage of the entries of the coefficient matrix are zero. If there is a pattern to the nonzero entries (i.e., tridiagonal systems), then an iterative process provides an efficient method for solving these large systems.

Sometimes the Jacobi method does not work. Let us experiment and see that a rearrangement of the original linear system can result in a system of iteration equations that will produce a divergent sequence of points.

Example 3.27. Let the linear system (1) be rearranged as follows:

$$-2x + y + 5z = -15$$
$$4x - 8y + z = -21$$
$$4x - y + z = 7.$$

These equations can be written in the form

$$x = \frac{15 + y + 5z}{2}$$
$$y = \frac{21 + 4x + z}{8}$$
$$z = 7 - 4x + y.$$

This suggests the following Jacobi iterative process:

$$x_{k+1} = \frac{15 + y_k + 5z_k}{3}$$
$$y_{k+1} = \frac{21 + 4x_k + z_k}{8}$$
$$z_{k+1} = 7 - 4x_k + y_k.$$

See that if we start with $P_0 = (x_0, y_0, z_0)$ then the iteration using (6) will diverge away from the solution $(2, 4, 3)$.

Substitute $x_0 = 1, y_0 = 2$, and $z_0 = 2$ into the right-hand side of each equation in (6) to obtain the new values x_1, y_1, and z_1:

$$x_1 = \frac{-15 + 2 + 10}{2} = -1.5$$
$$y_1 = \frac{21 + 4 + 2}{8} = 3.375$$
$$z_1 = 7 - 4 + 2 = 5.00$$

The new point $P_1 = (-1.5, 3.375, 5.00)$ is farther away from the solution $(2, 4, 3)$ than P_0. Iteration using the equations in (6) produces a divergent sequence (see Table 2.3).

Table 3.3 Divergent Jacobi iteration for the Linear System (4)

k	x_k	y_k	z_k
0	1.0	2.0	2.0
1	1.5	3.375	5.0
2	6.6875	2.5	16.375
3	34.6875	8.015625	17.25
4	46.617188	17.8125	123.73438
5	307.929688	36.150391	211.28125
6	502.62793	124.929688	1202.56836
$\vdots$	$\vdots$	$\vdots$	$\vdots$

2.4.2 Gauss-Seidel Iteration

Sometimes the convergence can be speeded up. Observe that the Jacobi iterative process (3) yields three sequences $\{x_k\}, \{y_k\}$, and $\{z_k\}$ that converge to 2, 4, and 3, respectively (see Table 3.2). It seems reasonable that $\{x_{k+1}\}$ could be used in place of $\{x_k\}$ in the computation of y_{k+1}. Similarly, x_{k+1} and y_{k+1} might be used in the computation of z_{k+1}. The next example shows what happens when this applied to the equations in Example 3.26.

Example 3.28. Consider the system of equations given in (1) and the Gauss-Seidel iterative process suggested by (2):

$$x_{k+1} = \frac{7 - y_k - z_k}{4}$$

$$y_{k+1} = \frac{21 + 4x_{k+1} + z_k}{8}$$

$$z_{k+1} = \frac{15 + 2x_{k+1} - y_{k+1}}{5}.$$

See that if we start with $P_0 = (x_0, y_0, z_0) = (1, 2, 3)$, then iteration using (7) will converge to the solution (2,4,3).

Substitute $y_0 = 2$ and $z_0 = 2$ into the first equation of (7) and obtain

$$x_1 = \frac{7 + 2 - 2}{4} = 1.75.$$

Then substitute $x_1 = 1.75$ and $z_0 = 2$ into the second equation and get

$$y_1 = \frac{21 + 4(1.75) + 2}{8} = 3.75.$$

Finally, substitute $x_1 = 1.75$ and $y_1 = 3.75$ into the third equation to get

$$z_1 = \frac{15 + 2(1.75) - 3.75}{5} = 2.95.$$

Table 3.2 Convergence Gauss-Seidel iteration for the Linear System (1)

k	x_k	y_k	z_k
0	1.0	2.0	2.0
1	1.75	3.75	2.95
2	1.95	3.96875	2.98625
3	1.995625	3.99609375	2.99903125
⋮	⋮	⋮	⋮
8	1.99999983	1.99999988	2.99999996
9	1.99999998	3.99999999	3.00000000
10	2.00000000	4.00000000	3.00000000

The new point $P_1 = (1.75, 3.75, 2.95)$ is closer to $(2, 4, 3)$.

In view of Example 3.26 and 3.27, it is necessary to have some criterion to determine whether the Jacobi iteration will converge. Hence we make the following definition.

Definition 3.6. A matrix A of dimension $N \times N$ is said to be strictly diagonally dominant provided that

$$|a_{kk}| > \sum_{j=1, j\neq k}^{N} |a_{k,j}| \quad \text{for } k = 1, 2, \ldots, N.$$

This means that in each row of the matrix the magnitude of the element on the main diagonal must exceed the sum of the magnitudes of all other elements in the row. The coefficient matrix of the linear system (1) in Example 3.26 is strictly diagonally dominant because

$$\begin{aligned} \text{In row 1:} &\quad |4| > |-1| + |1| \\ \text{In row 2:} &\quad |-8| > |4| + |1| \\ \text{In row 3:} &\quad |5| > |-2| + |1|. \end{aligned}$$

All the rows satisfy relation (8) in Definition 3.6; therefore, the coefficient matrix A for the linear system (1) strictly diagonally dominant.

The coefficient matrix A of the linear system (4) in Example 3.27 is not strictly diagonally dominant because

$$\begin{aligned} \text{In row 1:} &\quad |-2| > |1| + |5| \\ \text{In row 2:} &\quad |-8| > |4| + |1| \\ \text{In row 3:} &\quad |1| > |4| + |-1|. \end{aligned}$$

Rows 1 and 3 do not satisfy relation (8) in Definition 3.6; therefore, the coefficient matrix A for the linear system (4) is not strictly diagonally dominant.

We now generalized the Jacobi and Gauss-Seidel iteration processes. Suppose that the given linear system is

$$\begin{array}{c} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1j}x_j + \cdots + a_{1N}x_N = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2j}x_j + \cdots + a_{2N}x_N = b_2 \\ \vdots \quad \vdots \quad \vdots \\ a_{j1}x_1 + a_{j2}x_2 + \cdots + a_{jj}x_j + \cdots + a_{jN}x_N = b_j \\ \vdots \quad \vdots \quad \vdots \\ a_{N1}x_1 + a_{N2}x_2 + \cdots + a_{Nj}x_j + \cdots + a_{NN}x_N = b_N. \end{array}$$

Let the kth point be $P_k = (x_1^{(k)}, x_2^{(k)}, \ldots, x_j^{(k)}, \ldots, x_N^{(k)})$; then the next point is $P_{k+1} = (x_1^{(k+1)}, x_2^{(k+1)}, \ldots, x_j^{(k+1)}, \ldots, x_N^{(k+1)})$. The superscript (k) on the coordinates of P_k enable us to identify the coordinates that belong to this point. The iteration formulas

use row j of (9) to solve for $x_j^{(k+1)}$ in terms of a linear combination of the previous values $x_1^{(k)}, x_2^{(k)}, \ldots, x_j^{(k)}, \ldots, x_N^{(k)}$:
Jacobi iteration:

$$x_j^{(k+1)} = \frac{b_j - a_{j1}x_1^{(k)} - \cdots - a_{jj-1}x_{j-1}^{(k)} - a_{jj+1}x_{j+1}^{(k)} - \cdots - a_{jN}x_N^{(k)}}{a_{jj}}$$

for $j = 1, 2, \ldots, N$.

Jacobi iteration uses all old coordinates to generate all new coordinates, whereas Gauss-Seidel iteration uses the new coordinates as they become available:
Gauss-Seidel Iteration:

$$x_j^{(k+1)} = \frac{b_j - a_{j1}x_1^{(k+1)} - \cdots - a_{jj-1}x_{j-1}^{(k+1)} - a_{jj+1}x_{j+1}^{(k)} - \cdots - a_{jN}x_N^{(k)}}{a_{jj}}$$

for $j = 1, 2, \ldots, N$.

The following theorem gives a sufficient condition for Jacobi iteration to converge

Theorem 3.15 (Jacobi Iteration). Suppose that A is a strictly diagonally dominant matrix. Then $AX = b$ has a unique solution $X = P$. Iteration using formula (10) will produce a sequence of vectors $\{P_k\}$ that will converge to P for any choice of the starting vector P_0.
Proof. the proof can be found in advanced texts on numerical analysis.

It can be proved that the Gauss-Seidel method will also converge when the matrix A strictly diagonally dominant. In many cases the Gauss-Seidel method will converge faster than the Jacobi method: hence it is usually preferred (compare Examples 3.26 and 3.28). It is important to understand the slight modification of formula (10) that has been made to obtain formula (11). In some cases the Jacobi method will converge even though the Gauss-Seidel method will not.

2.4.3 Convergence

A measure of the closeness between vectors is needed so that we can determine if $\{P_k\}$ is converging to P. The Euclidean distance (see Section 3.1) between $P = (x_1, x_2, \ldots, x_n)$ and $Q = (y_1, y_2, \ldots, y_N)$ is

$$\|P - Q\| = \left(\sum_{j=1}^{N}(x_j - y_j)^2\right)^{1/2}.$$

Its disadvantage is that it requires considerable computing effort. Hence we introduce a different norm, $\|X\|_1$:

$$\|X\|_1 = \sum_{j=1}^{N} |x_j|.$$

The following result ensures that $\|X\|_1$ has the mathematical structure of a metric and hence is suitable to use as a generalized "distance formula." From the study of linear algebra we know that on a finite-dimensional vector space all norms are equivalent; that is, if two vectors are close in the $\| * \|_1$ norm, then they are also close in the Euclidean norm $\| * \|$.

Theorem 3.16. Let X and Y be N-dimensional vectors and c be a scalar. Then the function $\|X\|_1$ has the following properties:

$$\begin{aligned}
&\|X\|_1 \geq 0. \\
&\|X\|_1 = 0 \quad \text{if and only if} \quad X = 0. \\
&\|cX\|_1 = |c|\|X\|_1. \\
&\|X + Y\|_1 \leq \|X\| + \|Y\|_1.
\end{aligned}$$

Proof. We prove (17) and leave the others as exercises. For each j, the triangle inequality for real number states that $|x_j + y_j| \leq |x_j| + |y_j|$. Summing these yields inequality (17):

$$|X + Y|_1 = \sum_{j=1}^{N} |x_j + y_j| \leq \sum_{j=1}^{N} |x_j| + \sum_{j=1}^{N} |y_j| = \|X\|_1 + \|Y\|_1.$$

The norm given by (13) can be used to define the distance between points.

Definition 3.7. Suppose that X and Y are two points in N-dimensional space. We define the distance X and Y in the $\| * \|_1$ norm as

$$\|X - Y\|_1 = \sum_{j=1}^{N} |x_j - y_j|.$$

Example 3.29. Determine the Euclidean distance and $\| * \|_1$ distance between the points $P = (2, 4, 3)$ and $(Q = (1.75, 3.75, 2.95)$.

The Euclidean distance is

$$\|P - Q\| = \left((2 - 1.75)^2 + (4 - 3.75)^2 + (3 - 2.95)^2\right)^{1/2} = 0.3570.$$

The $\| * \|_1$ distance is

$$\|P - Q\|_1 = |2 - 1.75| + |4 - 3.75| + |3 - 2.95| = 0.55.$$

The $\| * \|_1$ is easier to compute and use for determining convergence in N-dimensional space.

The MATLAB command `A(j,[1:j-1,j+1:N])` is used in Program 3.4. This efficiently selects all elements in the jth row of A, except the element in the jth column (i.e, `A(j,j)`). This notation is used to simplify the Jacobi iteration (100 step in Program 3.4.

In both Program 3.4 and 3.5 we have used the MATLAB command norm, which is the Euclidean norm. The $\| * \|_1$ can also be used and the reader is encouraged to check the Help menu in MATLAB or one of the reference works for information on the norm command.