Cambridge University Press & Assessment
978-1-009-21419-3 — Computing in Scilab
Chetana Jain
Excerpt
More Information

1

Matrices and Vector Spaces

LEARNING OBJECTIVES

After going through this chapter, the reader should be able to

- ◊ Create different types of vectors and matrices in Scilab.
- ◊ Perform arithmetic operations on vectors and matrices.
- ◊ Solve linear and vector algebra in Scilab
- ◊ Construct and solve mathematical expressions that involve vectors, matrices and a system of linear equations.
- ◊ Solve advanced problems in physics using matrices.
- ◊ Construct differential and Laplace operators in Scilab.
- ◊ Determine the wave function of stationary states using the Hermitian differential operator.

1.1 Introduction

Matrices are often aptly described as key to solve everything in the scientific world. This chapter expounds the usefulness of vectors and matrices that occur in many kinds of problems across the disciplines. They are used to study innumerable physical phenomena such as, motion of rigid bodies, eigen states of a quantum mechanical system, electrical networks and coordinate system conversion.

In Scilab, matrix computation forms the basis of all calculations. This chapter recapitulates the basic Scilab rules that have to be followed for creating and editing matrices. It also summarizes the arithmetic operations that can be performed on matrices. *Section 1.2* gives an overview on different ways of generating a matrix and its elements. Some special types of matrices such as row/column vector, diagonal matrix, identity matrix and triangular matrices have been introduced in *Section 1.3*. Matrix operations such as row/column operation, conjugation, scalar/vector multiplication and division have been explained in *Section 1.4*. The laws of vector algebra have been outlined in *Section 1.5*. Some interesting examples of applications and use of matrices in physical sciences have been discussed in *Section 1.6*.

1.2 Creation of a Matrix

Matrices are rectangular arrangements of 'm' rows and 'n' columns; an arrangement of m rows and n columns is called an ($m \times n$) matrix. If it contains only one row or only one column, then it is called a vector. There are several ways of defining vectors and matrices in Scilab. Some of them have been explained as follows.

1. The elements of a matrix are defined by writing them inside a square bracket, such that the elements of a row are separated by a comma or a white space. The elements of consecutive rows are separated by a semi-colon.
2. The elements of a matrix can be of several types and have been listed in *Table 1.1*. As can be seen in this table,
    a. The elements can be real numbers.
    b. The elements can be complex numbers. The complex number consists of a real part and/or an imaginary part.
    c. The elements can be rational numbers, which are defined using the '`rlist`' command of Scilab.
    d. The elements can be random numbers, which are generated using the random number generator, '`rand`' command of Scilab. As can be seen in the table, it is also possible to format the number of significant digits in the random numbers.
    e. The elements can be character strings.
    f. The elements can be polynomials. In Scilab, '`%z`' is used as a variable for defining polynomials. As explained in the table, it is also possible to represent a polynomial by other variables using the '`poly(0,"variable")`'command.
    g. All the elements can be made equal to zero.
    h. All the elements can be made equal to ones.
    i. Elements of the matrix can follow a certain progression rule.

CAMBRIDGE
Cambridge University Press & Assessment
978-1-009-21419-3 — Computing in Scilab
Chetana Jain
Excerpt
More Information

Table 1.1 Creation of Matrices

S. No. (Method)	Scilab Command	Output Matrix
2 (a)	A = [1,2;3,4] Or A = [1 2;3 4]	A = 1. 2. 3. 4.
2 (b)	A = [%i 2 ; 3 4*%i]	A = i 2. 3. 4.i
2 (c)	Num = [1 1 ; 1 1]; Den = [1 2 ; 3 4]; A = rlist (Num,Den,[])	A = 1 1 - - 1 2 1 1 - - 3 4
2 (d)	A = rand(2,2)	A = 0.6856896 0.8415518 0.1531217 0.4062025
	format(5); A = rand(2,2)	A = 0.68 0.84 0.15 0.40
2 (e)	A = ["This" "is" ; "SciLab" "Program"]	A = !This is ! ! ! !SciLab Program !
2 (f)	A = [3 5+3*%z ; 7+%z^2 2+4*%z^2]	A = 3 5 + 3z $7 + z^2$ $2 + 4z^2$
	p = poly(0,'p'); A = [3*p 5+3*p ; 7+p^2 2+4*p^2]	A = 3p 5 + 3p $7 + p^2$ $2 + 4p^2$
	p = poly(0,'p'); A = [7 3*p ; 1/p^3 1/(1+2*p)]	A = 7 3p – --- 1 1 1 1 – ------ p^3 1 + 2p

S. No. (Method)	Scilab Command	Output Matrix
2 (g)	`A = zeros(2 , 2)` (A matrix of size 2 × 2 and all the elements are equal to zero)	A = 0. 0. 0. 0.
2 (h)	`A = ones(2 , 2)` (A matrix of size 2 × 2 and all the elements are equal to one)	A = 1. 1. 1. 1.
2 (i)	A = [1:4]	A = 1. 2. 3. 4.
	A = [1:2:10]	A = 1. 3. 5. 7. 9.

3. Elements of a matrix can also be defined from the console by writing the following Scilab program.

```
rows_num = input("How many rows?")
columns_num = input("How many columns?")
for i = 1:rows_num;
    for j=1:columns_num;
      A(i,j)= input("Enter the elements");
    end
end
```

1.3 Nature of the Matrix

It is straightforward to define different types of matrices in Scilab. The commands for some of them have been listed in *Table 1.2*.

Table 1.2 Types of Matrices

Type of Matrix	Scilab Command	Output Matrix
Row matrix (1 × 3)	A = [5 3 8]	A = 5. 3. 8.
Column matrix (3 × 1)	A = [5; 3; 8]	A = 5. 3. 8.

Type of Matrix	Scilab Command	Output Matrix
Diagonal matrix (2 × 2)	A = diag ([1 2])	A = 1. 0. 0. 2.
	A = diag ([rand(),rand()])	A = 0.349361 0. 0. 0.387377
Identity matrix (2× 2)	A = eye (2,2)	A = 1. 0. 0. 1.
Triangular matrix (Lower triangle)	A = tril([1 2 3 ; 4 5 6 ; 7 8 9])	A = 1. 0. 0. 4. 5. 0. 7. 8. 9.
	A = tril ([1 2 3 ; 4 5 6 ; 7 8 9],[-1])	A = 0. 0. 0. 4. 0. 0. 7. 8. 0.
	A = tril ([1 2 3 ; 4 5 6 ; 7 8 9],[1])	A = 1. 2. 0. 4. 5. 6. 7. 8. 9.
Triangular matrix (Upper triangle)	A = triu ([1 2 3 ; 4 5 6 ; 7 8 9])	A = 1. 2. 3. 0. 5. 6. 0. 0. 9.

1.4 Matrix Operation

The elementary matrix operations in Scilab have been listed in *Table 1.3*.

Table 1.3 Matrix Operations

Matrix Operation	Scilab Command	Output Matrix
Transpose of a row matrix	A = [7 9 1] B = A.′	B = 7. 9. 1.
Transpose of a column matrix	A = [3 ; 7 ; 2] B = A.′	B = 3. 7. 2.

Cambridge University Press & Assessment
978-1-009-21419-3 — Computing in Scilab
Chetana Jain
Excerpt
More Information

Matrix Operation	Scilab Command	Output Matrix
Conjugate of a matrix	A = [1+2*%i 5*%i ; 11*%i 4-%i] B = conj (A)	A = 1.+2.i 5.i 11.i 4. - i B = 1.-2.i - 5.i - 11.i 4.+i
Conjugate transpose of a matrix	A = [1+2*%i 5*%i ; 11*%i 4-%i] B = A′	A = 1.+2.i 5.i 11.i 4. - i B = 1. - 2.i - 11.i - 5.i 4. + i
Interchange of first and second row	A=[1 2 3;4 5 6;7 8 9] A([1,2],:) = A([2,1],:)	A = 1. 2. 3. 4. 5. 6. 7. 8. 9. A = 4. 5. 6. 1. 2. 3. 7. 8. 9.
Interchange of first and second column	A = [1 2 3; 4 5 6; 7 8 9] A(:,[1,2]) = A(:,[2,1])	A = 1. 2. 3. 4. 5. 6. 7. 8. 9. A = 2. 1. 3. 5. 4. 6. 8. 7. 9.
Square of matrix	A = [2 1; 4 3] B = A^2	B = 8. 5. 20. 13.
	A = [2 1;4 3] B = A**2	B = 8. 5. 20. 13.
Square of elements of the matrix	A = [2 1;4 3] B = A.^2	B = 4. 1. 16. 9.
Square root of elements of the matrix	A = [2 1;4 3] B = sqrt (A)	B = 1.414 1. 2. 1.732
Product of all the elements of the matrix	A = [2 1;4 3] B = prod (A)	B = 24

Cambridge University Press & Assessment
978-1-009-21419-3 — Computing in Scilab
Chetana Jain
Excerpt
More Information

Matrix Operation	Scilab Command	Output Matrix
Sum of two matrices	A = [1 3 ; 5 2] B = [4 7 ; 6 4] C = A + B	A = 1. 3. 5. 2. B = 4. 7. 6. 4. C = 5. 10. 11. 6.
Difference of two matrices	A = [1 3 ; 5 2] B = [4 7 ; 6 4] C = B - A	A = 1. 3. 5. 2. B = 4. 7. 6. 4. C = 3. 4. 1. 2.
Product of two matrices	A = [1 3 ; 5 2] B = [4 7 ; 6 4] C = A * B D = B * A	A = 1. 3. 5. 2. B = 4. 7. 6. 4. C = 22. 19. 32. 43. D = 39. 26. 26. 26.
Product of two matrices (element wise)	A = [1 3 ; 5 2] B = [4 7 ; 6 4] C = A.*B D = B.*A	A = 1. 3. 5. 2. B = 4. 7. 6. 4. C = 4. 21. 30. 8. D = 4. 21. 30. 8.
Matrix division	A = [7 9 ; 4 6] B = [3 1 ; 5 9] C = B/A	A = 7. 9. 4. 6. B = 3. 1. 5. 9. C = 2.333 - 3.333 - 1. 3.

Cambridge University Press & Assessment
978-1-009-21419-3 — Computing in Scilab
Chetana Jain
Excerpt
More Information

Matrix Operation	Scilab Command	Output Matrix
Matrix division (element wise)	A = [7 9 ; 4 6] B = [3 1 ; 5 9] C = B./A	A = 7. 9. 4. 6. B = 3. 1. 5. 9. C = 0.428 0.111 1.25 1.5
Trace of the matrix	A=[7 3 3 ; 4 7 6 ; 7 8 1] B = trace(A)	A = 7. 3. 3. 4. 7. 6. 7. 8. 1. B = 15.
	A=[7 3 3;4 7 6;7 8 1] B = sum(diag(A)) (Sum of diagonal elements)	A = 7. 3. 3. 4. 7. 6. 7. 8. 1. B = 15.
Determinant of the matrix	A=[1 0 0;0 2 0;0 0 3] B = det(A)	A = 1. 0. 0. 0. 2. 0. 0. 0. 3. B = 6.
Eigen value and Eigen vector of the matrix	A=[1 0 0;0 2 0;0 0 3] B = spec(A)	A = 1. 0. 0. 0. 2. 0. 0. 0. 3. B = 1. 2. 3.
	[B,C] = spec(A) (Matrix B contains Eigen vectors) (Matrix C contains Eigen values along the diagonal)	A = 1. 0. 0. 0. 2. 0. 0. 0. 3. B = 1 0. 0. 0. 1. 0. 0. 0. 1. C = 1. 0. 0. 0. 2. 0. 0. 0. 3.

Matrix Operation	Scilab Command	Output Matrix
Matrix inverse	A = [2 1 ; 4 3] B = inv(A)	A = 2. 1. 4. 3. B = 1.5 - 0.5 - 2. 1.

1.5 Vector Algebra

This section describes the use of Scilab for applications in vector algebra. The most commonly used laws of vector algebra have been listed in *Table 1.4*.

Table 1.4 Vector Algebra

Vector Algebra	Scilab Command	Output
Sum of elements of a vector	A = [2 1 3]; B = sum (A)	B = 6
Product of elements of a vector	A = [2 3 4]; B = prod (A)	B = 24
Maximum value in a vector	A = [2 3 1]; B = max (A)	B = 3
Minimum value in a vector	A = [2 3 1]; B = min (A)	B = 1
Number of elements	A = [2 1 4]; B = length (A)	B = 3
Value of second element of a vector	A = [2 3 1]; B = A (2)	B = 3
Magnitude of a vector	A = [2 1 4]; B = norm (A)	B = 4.5825757
Unit vector	A = [2 1 4]; B = A/norm (A)	B = 0.4364358 0.2182179 0.8728716

Vector Algebra	Scilab Command	Output
Projection of a vector B over vector A	A = [2 1 4]; B = [5 4 7]; C = A*B′/norm (A)	C = 9.1651514
Dot product of two vectors	A = [2 1 4]; B = [5 4 7]; C = sum (A.*B)	C = 42
Cross product of two vectors	A = [2; 1; 4]; B = [5; 4; 7]; C = cross (A,B)	C = - 9. 6. 3.

1.6 Applications

1.6.1 Coordinate conversion (Cartesian to cylindrical coordinate system)

The Cartesian coordinates are the simplest and, as far as calculations are concerned, the most sought after system of coordinates. However, there are several physical phenomena and systems that involve rotational symmetry about an axis. For example, flow of current through a long straight wire and flow of water through a long straight pipe having circular cross-section. In these problems, the use of the Cartesian coordinate system becomes tedious and it is preferable to perform the calculations in the cylindrical coordinate system. As will be explained in this section, matrices are easy and useful for converting the coordinates of a point from one coordinate system to another coordinate system.

Suppose the coordinates of a point (P) in Cartesian system are denoted by (x,y,z). The corresponding cylindrical coordinates are denoted by (r, θ, z). Here,

- 'r' is the perpendicular distance of point P from the z-axis.
- 'θ' is the azimuthal angle. It is the angle between the x-axis and the line joining the origin with the projection of point P on the x - y plane.
- 'z' is the perpendicular distance of point P from the x - y plane

The Cartesian and cylindrical coordinates of point P are related by *Eqn. 1.1–1.6*.

- Cartesian → Cylindrical

$$r = (x^2 + y^2)^{1/2} \tag{1.1}$$