Chapter 2: Numeric, Cell, and Structure Arrays

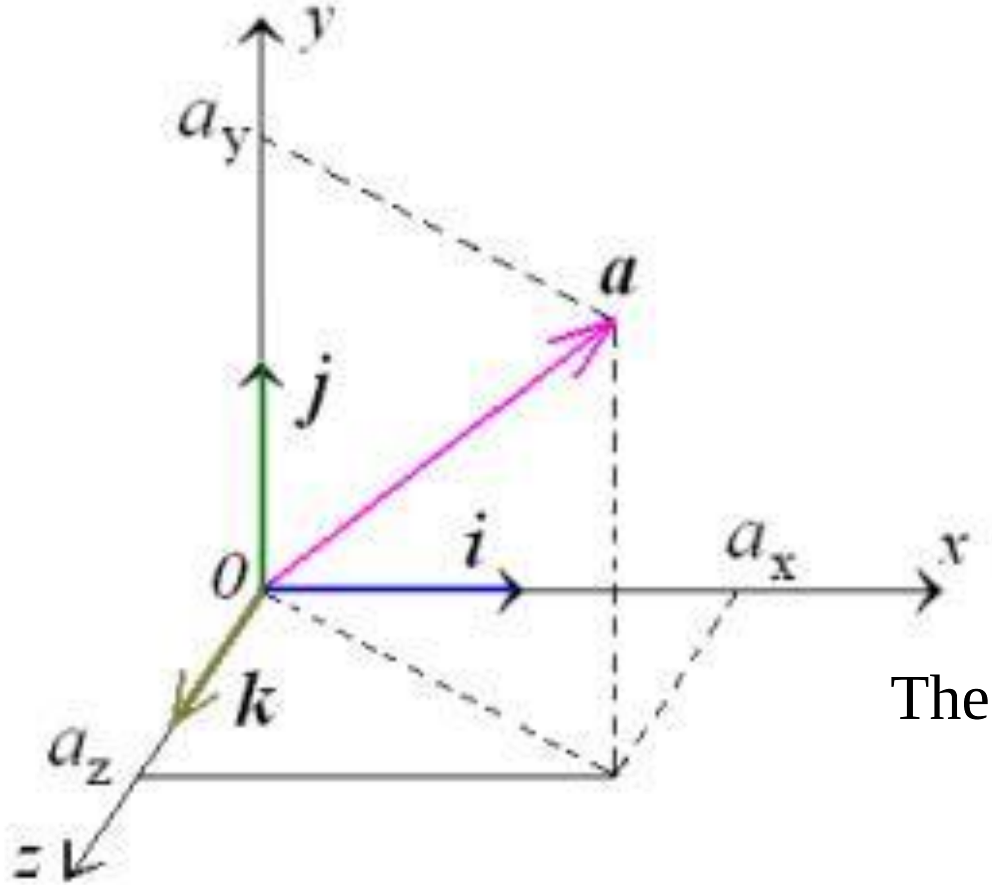
Topics Covered:

- Vectors
  - Definition
  - Addition
  - Multiplication
    - Scalar, Dot, Cross
- Matrices
  - Row, Column, Square
  - Transpose
  - Addition
  - Multiplication
    - Scalar-Matrix, Matrix-Matrix
  - Element-by-Element Operations
    - Multiplication, Division, Exponentiation, Vectorized Functions
  - Array Addressing

Vectors

A **Scalar** is a number that has magnitude but no direction (Temperature reading on a thermometer, pressure reading on a pressure gage)

A **Vector** has both magnitude and direction. Vectors are used by engineers to represent position, velocity, acceleration, force, torque, angular velocity, angular acceleration, linear momentum, angular momentum, heat flux, magnetic flux, etc.



The **Vector** $\vec{a}$ is composed of three Cartesian **Components**:

$$\vec{a} = (a_x)\hat{\imath} + (a_y)\hat{\jmath} + (a_z)\hat{k}$$

$\hat{\imath}, \hat{\jmath},$ and $\hat{k}$ are **Unit Vectors** in the x, y, z directions.

The **Length** of a **Unit Vector** is one.

The **Length** of the **Vector** $\vec{a}$ is called its **Magnitude**:

$$|\vec{a}| = \sqrt{a_x^2 + a_y^2 + a_z^2}$$

Vectors are represented in MATLAB as:

$$\vec{a} = [a_x, a_y, a_z] \text{ or } [a_x \quad a_y \quad a_z]$$

Open up a new MATLAB Script File:

Editor - C:\Users\scott\Desktop\chapter2.m

chapter2.m

```
clc
clear
a = [3, 7, 15]
Number_of_Elements = length(a)
Magnitude_of_the_Vector = norm(a)
Add_the_Elements = sum(a)
Maximum_Element = max(a)
```

Command Window

```
a =

     3     7    15


Number_of_Elements =

     3


Magnitude_of_the_Vector =

   16.8226


Add_the_Elements =

    25


Maximum_Element =

    15
```

<u>Vector Addition:</u> Add the corresponding **Components** of the two **Vectors**:

$$\vec{a} + \vec{b} = [a_x, a_y, a_z] + [b_x, b_y, b_z]$$

$$\vec{a} + \vec{b} = [(a_x + b_x), (a_y + b_y), (a_z + b_z)]$$

$$\vec{a} - \vec{b} = [(a_x - b_x), (a_y - b_y), (a_z - b_z)]$$

Modify and run the Script File as follows:

Editor - C:\Users\scott\Desktop\cha...

chapter2.m

```
clc
clear
a = [3, 7, 15]
b = [1, 5, -9]
x = a + b
y = a - b
```

Command Window

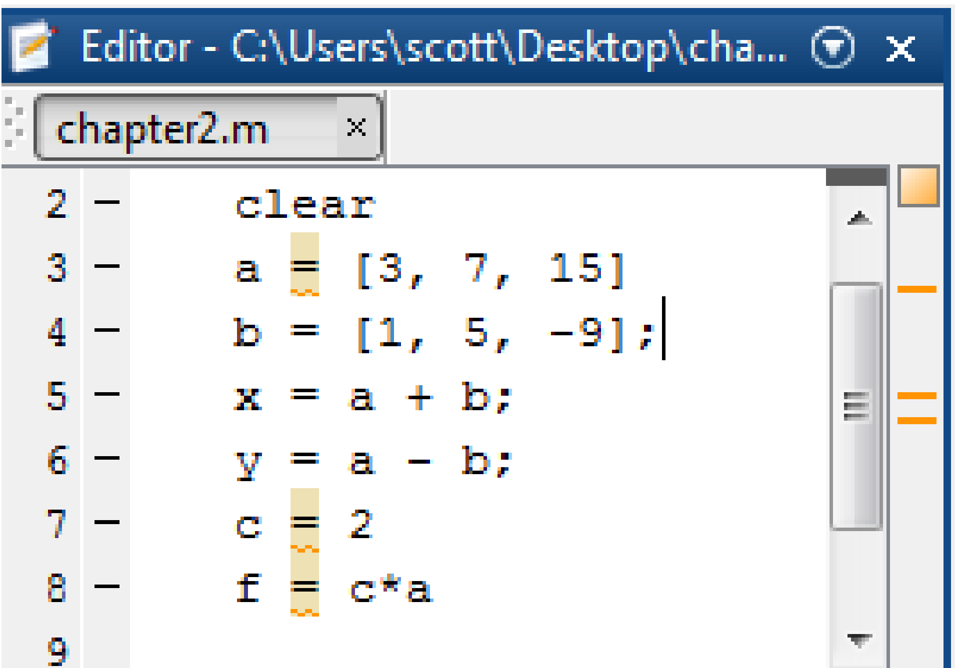
```
a =

     3     7    15


b =

     1     5    -9


x =

     4    12     6


y =

     2     2    24
```

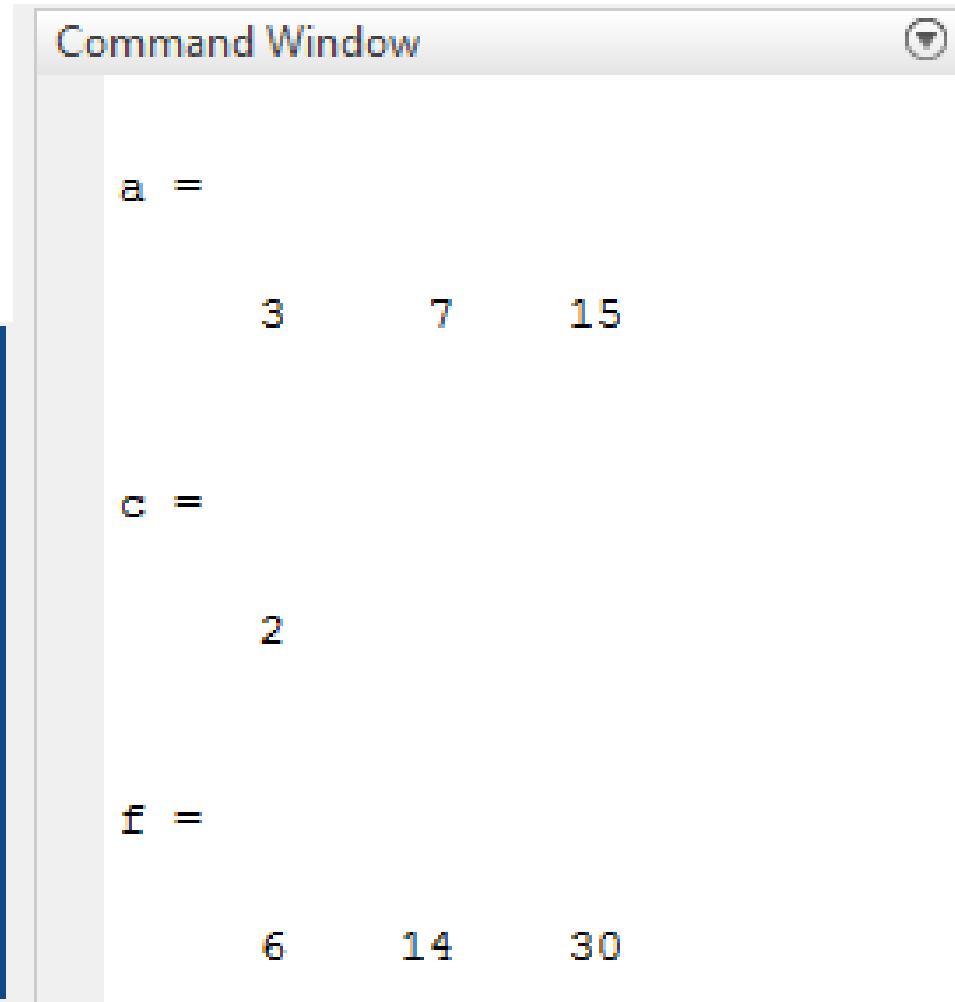
Vector Multiplication

Scalar × **Vector**: Distribute the **Scalar** to each **Component** of the **Vector**:

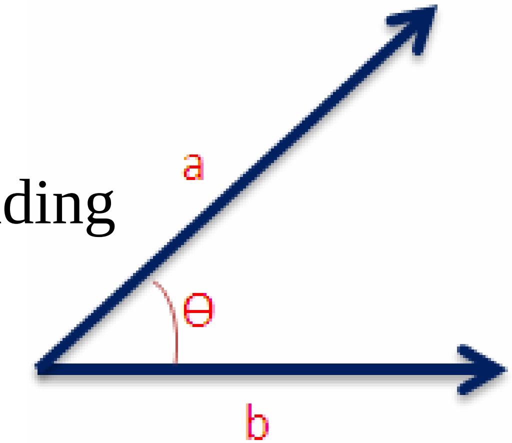
$$c\vec{a} = [ca_x, ca_y, ca_z]$$

Modify and run the Script File as follows:





Vector Dot Product



Vector · Vector: Sum of the products of the corresponding **Components** of the two **Vectors**.

$$\vec{a} \cdot \vec{b} = |\vec{a}| \times |\vec{b}| \times \cos\theta = \textbf{Scalar}$$

$$\vec{a} \cdot \vec{b} = \vec{b} \cdot \vec{a} = a_x b_x + a_y b_y + a_z b_z$$

Modify and run the Script File as follows:

Editor - C:\Users\scott\Desktop\cha...

chapter2.m

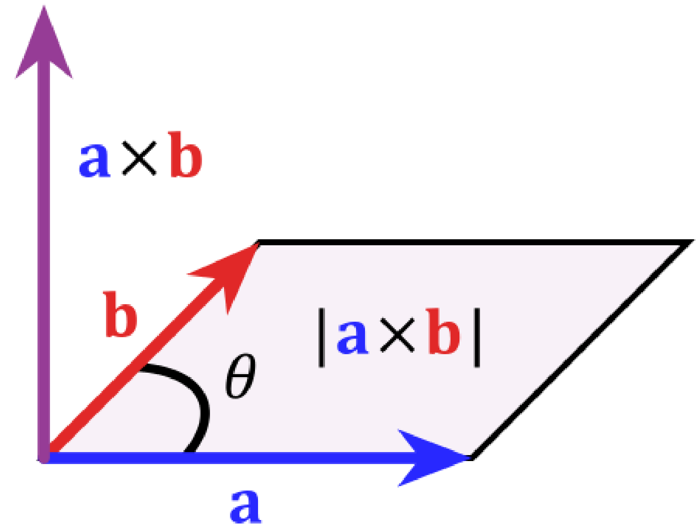
```
a = [3, 7, 15]
b = [1, 5, -9]
x = a + b;
y = a - b;
c = 2;
f = c*a;
g = dot(a,b)
```

Command Window

```
a =

     3     7    15


b =

     1     5    -9


g =

   -97
```

Vector Cross Product

Vector × **Vector**: Results in a Vector **Normal** (Perpendicular) to the plane defined by the two vectors being crossed.



$$\vec{a} \times \vec{b} = (|\vec{a}||\vec{b}|\sin\theta)\hat{n} = \textbf{Vector}$$

$\hat{n}$ is the **Unit Vector Normal** to the **Plane**.

$$\vec{a} \times \vec{b} = -(\vec{b} \times \vec{a}) = \begin{bmatrix} \hat{\imath} & \hat{\jmath} & \hat{k} \\ a_x & a_y & a_z \\ b_x & b_y & b_z \end{bmatrix}$$

Determinant Method

$$\vec{a} \times \vec{b} = (a_y b_z - a_z b_y)\hat{\imath} - (a_x b_z - a_z b_x)\hat{\jmath} + (a_x b_y - a_y b_x)\hat{k}$$

Modify and run the Script File as follows:

Editor - C:\Users\scott\Desktop\cha...

chapter2.m

```
clc
clear
a = [3, 7, 15]
b = [1, 5, -9]
x = a + b;
y = a - b;
c = 2;
f = c*a;
g = dot(a,b);
h = cross(a,b)
m = cross(b,a)
```

Command Window

```
a =

     3     7    15


b =

     1     5    -9


h =

  -138    42     8


m =

   138   -42    -8
```

Matrices

A **Matrix** is a rectangular **Array** of numbers arranged in **Rows** and **Columns**. The individual numbers in a **Matrix** are called **Elements**.

$$\mathbf{A} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}$$

Row 1

Column 1

$A_{\text{Row}-\text{Column}}$: A_{23} = Number in the Second Row, Third Column

In MATLAB, use a **Comma** to separate **Columns**; use a **Semi-Colon** to separate **Rows**:

$$\mathbf{A} = [A_{11}, A_{12}, A_{13}; A_{21}, A_{22}, A_{23}; A_{31}, A_{32}, A_{33}]$$

Modify and run the Script File as follows:

Editor - C:\Users\scott\Desktop\chapter2.m*

chapter2.m*

```
clc
clear
a = [3, 7, 15];
b = [1, 5, -9];
x = a + b;
y = a - b;
c = 2;
f = c*a;
g = dot(a,b);
h = cross(a,b);
m = cross(b,a);

A = [1, 2, 3; 4, 5, 6; 7, 8, 9]
```

Command Window

```
A =

     1     2     3
     4     5     6
     7     8     9
```

Row Vector: A **Matrix** with one **Row**, multiple **Columns**

$$\mathbf{B} = [B_{11}, B_{12}, B_{13}]$$

Column Vector: A **Matrix** with one **Column**, multiple **Rows**

$$\mathbf{C} = \begin{bmatrix} C_{11} \\ C_{21} \\ C_{31} \end{bmatrix}$$

In MATLAB, create a **Column Vector** by using **Semi-Colons** to separate **Rows**

$$\mathbf{C} = [C_{11}; C_{12}; C_{13}]$$

Square Matrix: Number of **Rows** = Number of **Columns**

$$\mathbf{D} = \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix}$$

A **Vector** can be **Transposed** by switching the **Rows** and **Columns**.

$$\mathbf{A} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \qquad \mathbf{A}^{\mathbf{T}} = \begin{bmatrix} A_{11} & A_{21} & A_{31} \\ A_{12} & A_{22} & A_{32} \\ A_{13} & A_{23} & A_{33} \end{bmatrix}$$

Modify and run the Script File as follows:

Editor - C:\Users\scott\Desktop\chapter2.m

chapter2.m

```
h = cross(a,b);
m = cross(b,a);

A = [1, 2, 3; 4, 5, 6; 7, 8, 9]
E = A'
```

Command Window

```
A =

     1     2     3
     4     5     6
     7     8     9


E =

     1     4     7
     2     5     8
     3     6     9
```

Matrix Addition: The **Size** of the two **Matrices** must be the same:

$(2 \times 2) + (2 \times 2) = (2 \times 2)$ (Two Rows × Two Columns)

$$\mathbf{D} + \mathbf{F} = \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix} + \begin{bmatrix} F_{11} & F_{12} \\ F_{21} & F_{22} \end{bmatrix}$$

$$\mathbf{D} + \mathbf{F} = \begin{bmatrix} (D_{11}+F_{11}) & (D_{12}+F_{12}) \\ (D_{21}+F_{21}) & (D_{22}+F_{22}) \end{bmatrix}$$

Modify and run the Script File as follows:

Editor - C:\Users\scott\Desktop\chapter2.m

chapter2.m

```
A = [1, 2, 3; 4, 5, 6; 7, 8, 9];
E = A';

D = [1 2; 3 4]
F = [5 6; 7 8]
G = D + F
```

Command Window

```
D =

     1     2
     3     4


F =

     5     6
     7     8


G =

     6     8
    10    12
```

Scalar × Matrix Multiplication: The **Scalar** is **Distributed** to all of the **Elements** of the **Matrix**:

$$\mathbf{cA} = \begin{bmatrix} cA_{11} & cA_{12} & cA_{13} \\ cA_{21} & cA_{22} & cA_{23} \\ cA_{31} & cA_{32} & cA_{33} \end{bmatrix}$$

Modify and run the Script File as follows:

Editor - C:\Users\scott\Desktop\chapter2.m

chapter2.m

```
A = [1, 2, 3; 4, 5, 6; 7, 8, 9]
E = A';

D = [1 2; 3 4];
F = [5 6; 7 8];
G = D + F;
c = 3
H = c*A
```

Command Window

```
A =

     1     2     3
     4     5     6
     7     8     9


c =

     3


H =

     3     6     9
    12    15    18
    21    24    27
```

Matrix × Matrix Multiplication: The **Size** of the two **Matrices Can Be the Same**: (2 × 2) × (2 × 2) = (2 × 2)

$$\mathbf{D} \times \mathbf{F} = \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix} \times \begin{bmatrix} F_{11} & F_{12} \\ F_{21} & F_{22} \end{bmatrix}$$

$$\mathbf{D} \times \mathbf{F} = \begin{bmatrix} (D_{11}F_{11} + D_{12}F_{21}) & (D_{11}F_{12} + D_{12}F_{22}) \\ (D_{21}F_{11} + D_{22}F_{21}) & (D_{21}F_{12} + D_{22}F_{22}) \end{bmatrix}$$

Modify and run the Script File as follows:

Editor - C:\Users\scott\Desktop\

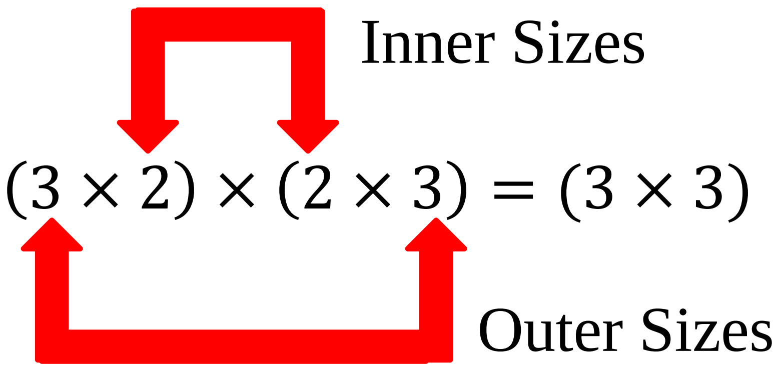
chapter2.m*

```
D = [1 2; 3 4]
F = [5 6; 7 8]
G = D + F;
c = 3;
H = c*A;
J = D*F
```

Command Window

```
D =

     1     2
     3     4


F =

     5     6
     7     8


J =

    19    22
    43    50
```

Matrix × Matrix Multiplication: The **Sizes** of the two **Matrices Can Be Different**: The **Inner Sizes Must** be the **Same**; the **Outer Sizes Can** be **Different**.



$$\mathbf{K} \times \mathbf{L} = \begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \\ K_{31} & K_{32} \end{bmatrix} \times \begin{bmatrix} L_{11} & L_{12} & L_{13} \\ L_{21} & L_{22} & L_{23} \end{bmatrix}$$

$$= \begin{bmatrix} (K_{11}L_{11} + K_{12}L_{21}) & (K_{11}L_{12} + K_{12}L_{22}) & (K_{11}L_{13} + K_{12}L_{23}) \\ (K_{21}L_{11} + K_{22}L_{21}) & (K_{21}L_{12} + K_{22}L_{22}) & (K_{21}L_{13} + K_{22}L_{23}) \\ (K_{31}L_{11} + K_{32}L_{21}) & (K_{31}L_{12} + K_{32}L_{22}) & (K_{31}L_{13} + K_{32}L_{23}) \end{bmatrix}$$

Modify and run the Script File as follows:

Editor - C:\Users\scott\Desktop\chapter2.m

chapter2.m

```
H = c*A;
J = D*F;

K = [1 2; 3 4; 5 6]
L = [7 8 9; 10 11 12]
M = K*L
```

Command Window

```
K =

     1     2
     3     4
     5     6


L =

     7     8     9
    10    11    12


M =

    27    30    33
    61    68    75
    95   106   117
```

What Happens when you Reverse the Order of Multiplication? Try it!

$$(2 \times 3) \times (3 \times 2) = (2 \times 2)$$

$$\mathbf{L} \times \mathbf{K} = \begin{bmatrix} L_{11} & L_{12} & L_{13} \\ L_{21} & L_{22} & L_{23} \end{bmatrix} \times \begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \\ K_{31} & K_{32} \end{bmatrix}$$

$$= \begin{bmatrix} (L_{11}K_{11} + L_{12}K_{21} + L_{13}K_{31}) & (L_{11}K_{12} + L_{12}K_{22} + L_{13}K_{32}) \\ (L_{21}K_{11} + L_{22}K_{21} + L_{23}K_{31}) & (L_{21}K_{12} + L_{22}K_{22} + L_{23}K_{32}) \end{bmatrix}$$

In General,

Inner Sizes Must be Equal

$$(n \times m) \times (m \times p) = (n \times p)$$

Outer Sizes Can Be Different

Another method to calculate the **Dot Product** of two vectors is to use **Matrix Multiplication**:

$$\mathbf{B} = [B_{11}, B_{12}, B_{13}]$$

$$\mathbf{C} = [C_{11}, C_{12}, C_{13}]$$

$$\mathbf{B} \cdot \mathbf{C} = [B_{11}\ B_{12}\ B_{13}] * \begin{bmatrix} C_{11} \\ C_{21} \\ C_{31} \end{bmatrix}$$

Verify this using MATLAB!

Let:

$$\mathbf{B} = [1, 2, 3]$$

$$\mathbf{C} = [4, 5, 6]$$

Element × Element Multiplication:

Element-by-Element Multiplication is often used in engineering calculations. It is defined only for arrays that have the same size.

$$\mathbf{D}.\times\ \mathbf{F} = \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix} .* \begin{bmatrix} F_{11} & F_{12} \\ F_{21} & F_{22} \end{bmatrix}$$

$$\mathbf{D}.\times\ \mathbf{F} = \begin{bmatrix} (D_{11}F_{11}) & (D_{12}F_{12}) \\ (D_{21}F_{21}) & (D_{22}F_{22}) \end{bmatrix}$$

Editor - C:\Users\scott\Desktop\c

chapter2.m

```
D = [1 2; 3 4]
F = [5 6; 7 8]

N = D.*F

```

Command Window

```
    61    68
    95   106


D =

     1     2
     3     4


F =

     5     6
     7     8


N =

     5    12
    21    32
```

Element × Element Division:

Element-by-Element Division is also defined only for arrays that have the same size.

$$\mathbf{D./F} = \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix} ./ \begin{bmatrix} F_{11} & F_{12} \\ F_{21} & F_{22} \end{bmatrix}$$

$$\mathbf{D./F} = \begin{bmatrix} (D_{11}/F_{11}) & (D_{12}/F_{12}) \\ (D_{21}/F_{21}) & (D_{22}/F_{22}) \end{bmatrix}$$

Editor - C:\Users\scott\Desktop\c

chapter2.m

```
D = [1 2; 3 4]
F = [5 6; 7 8]
N = D.*F;

P = D./F

```

Command Window

```
D =

     1     2
     3     4


F =

     5     6
     7     8


P =

    0.2000    0.3333
    0.4286    0.5000
```

Element × Element Exponentiation:

Element-by-Element Exponentiation is also defined only for arrays that have the same size.

$$\mathbf{D.\,^\wedge F} = \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix} .^\wedge \begin{bmatrix} F_{11} & F_{12} \\ F_{21} & F_{22} \end{bmatrix}$$

$$\mathbf{D.\,^\wedge F} = \begin{bmatrix} (D_{11}{}^\wedge F_{11}) & (D_{12}{}^\wedge F_{12}) \\ (D_{21}{}^\wedge F_{21}) & (D_{22}{}^\wedge F_{22}) \end{bmatrix}$$

Editor - C:\Users\scott\Desktop

chapter2.m

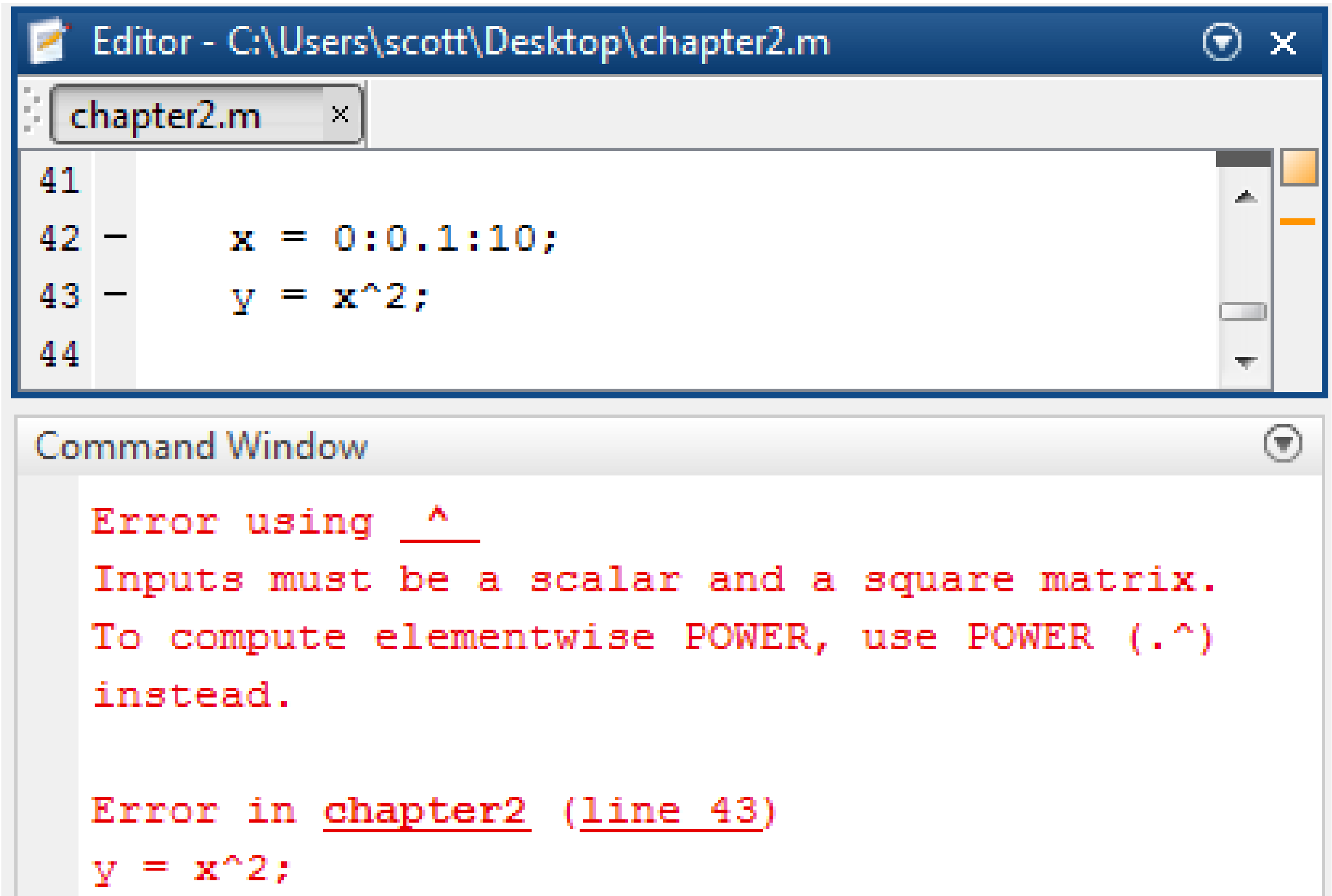
```
D = [1 2; 3 4]
F = [5 6; 7 8]
N = D.*F;
P = D./F;
Q = D.^F
```

Command Window

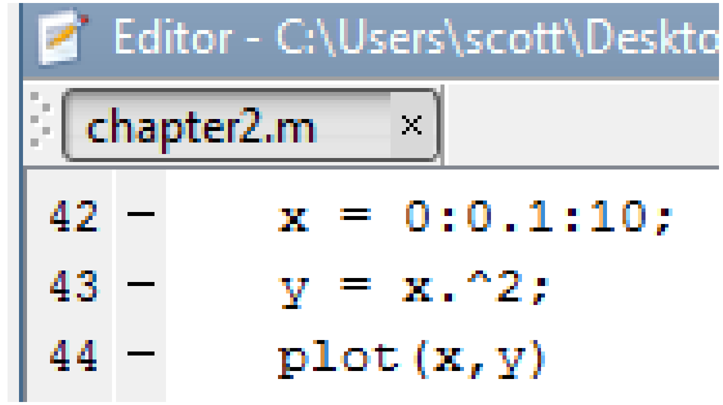
```
D =

     1     2
     3     4


F =

     5     6
     7     8


Q =

           1          64
        2187       65536
```

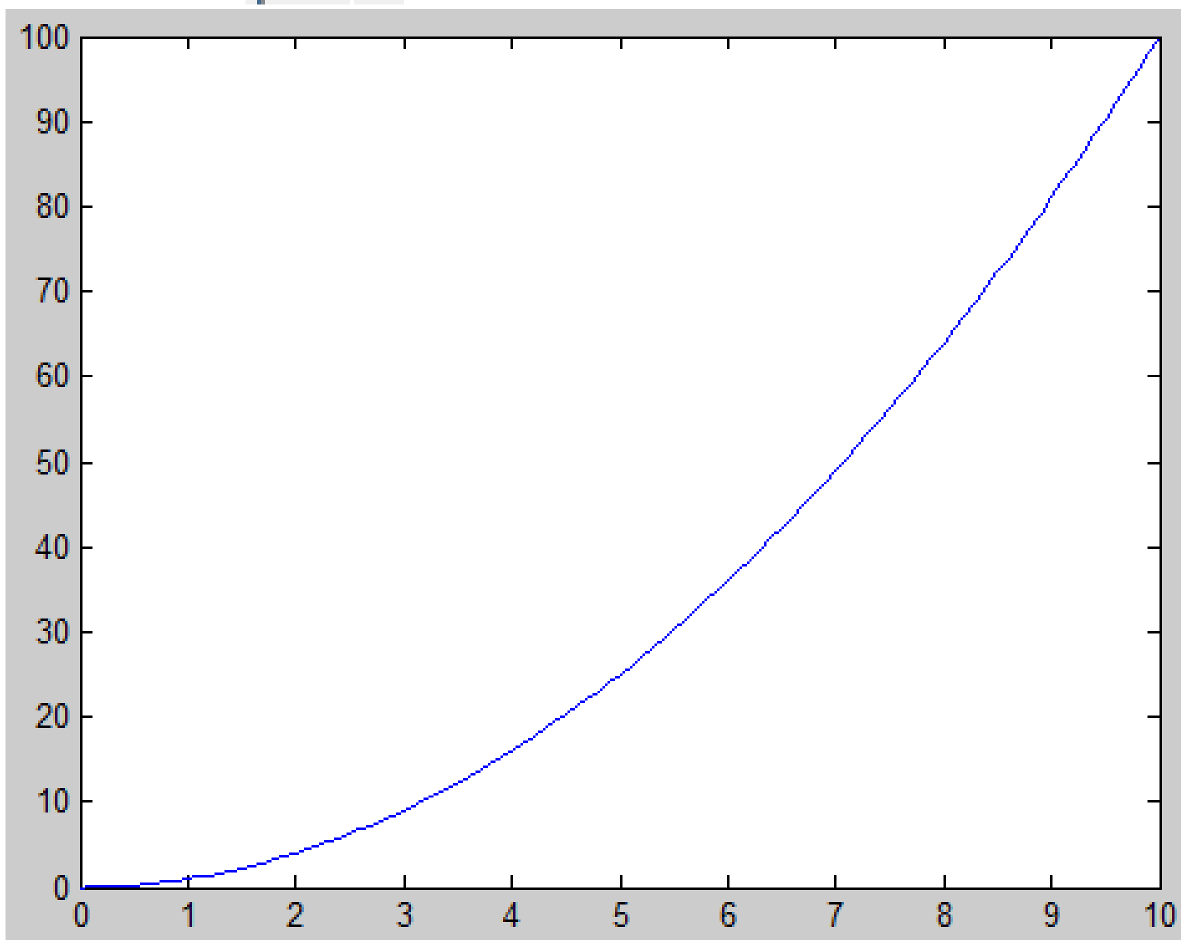
Vectorized Functions:

When you use functions like sin(*x*), atan(*x*) and exp(*x*) on vectors, the results of the functions become vectors also. These are called **Vectorized Functions**. When multiplying or dividing these functions, you must use **Element-by-Element Operations**.

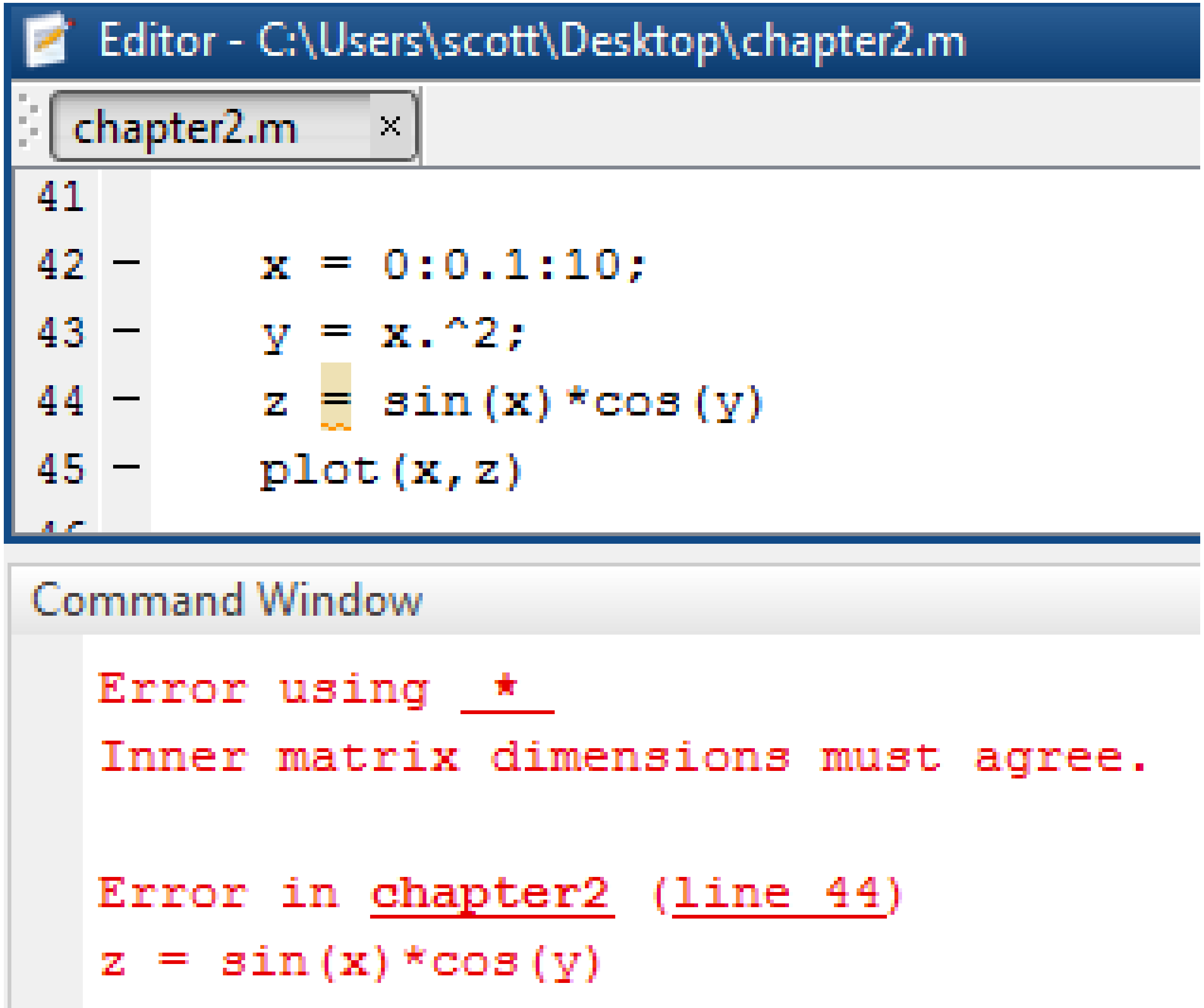


Vectorized Functions:

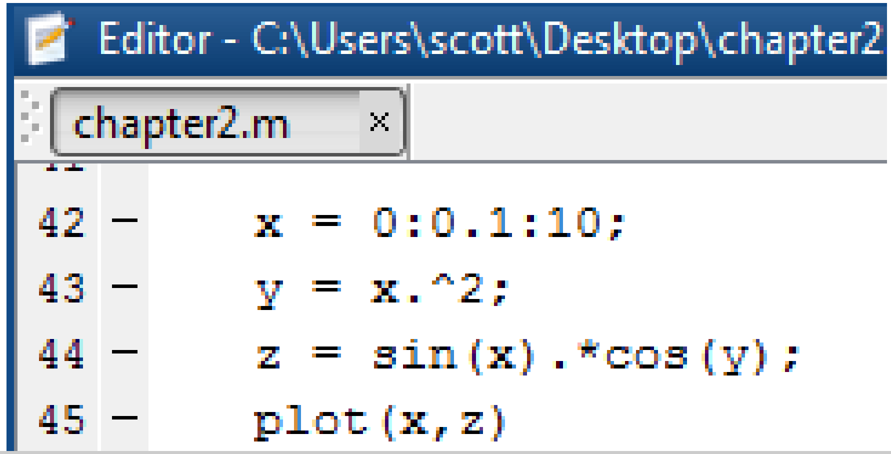


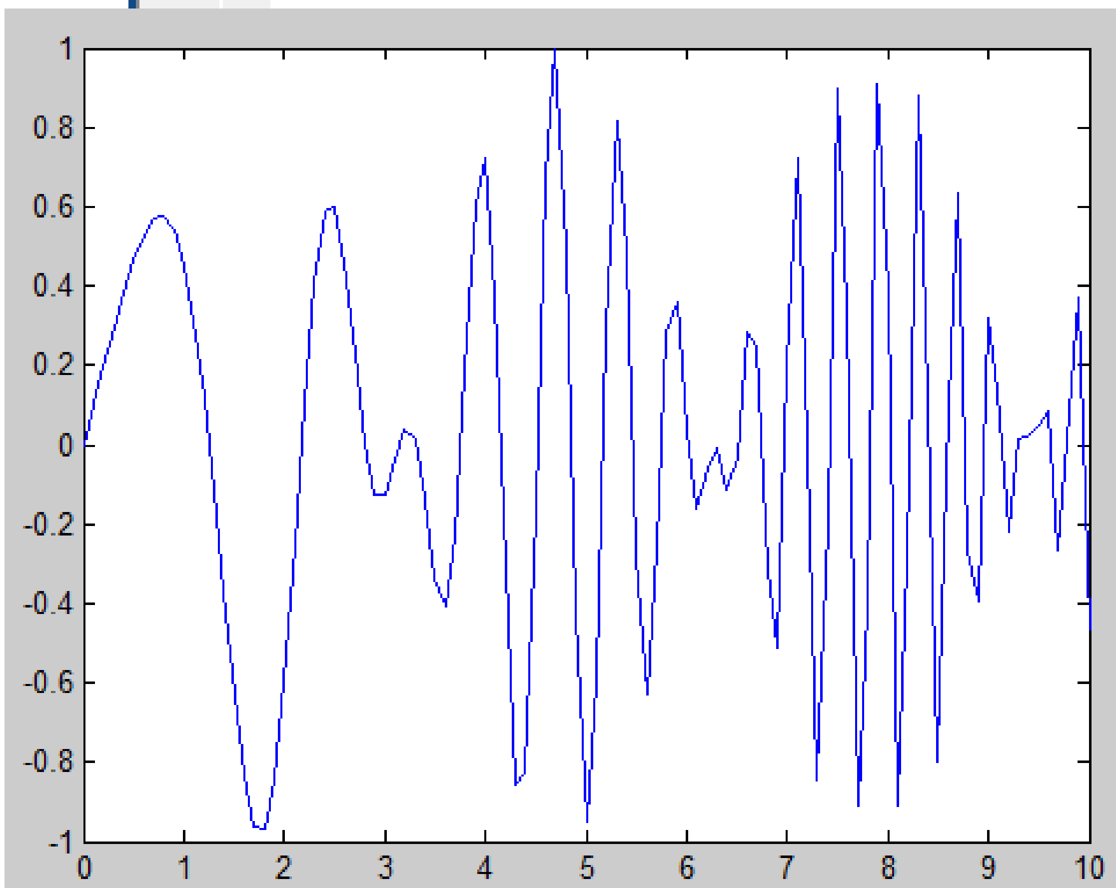


Vectorized Functions:



Vectorized Functions:





Vectorized Functions:

Editor - C:\Users\scott\Desktop\chapter2.m

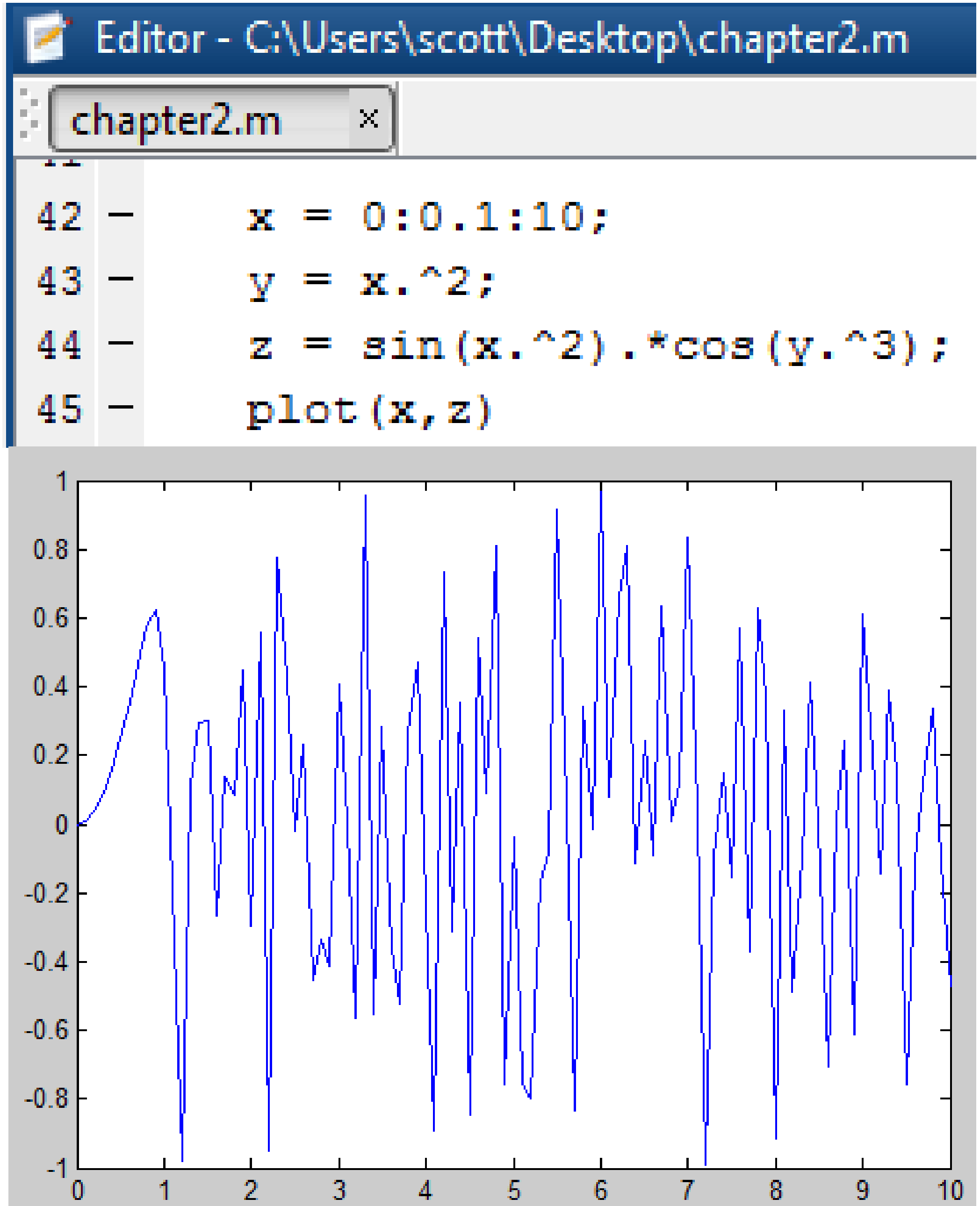
chapter2.m

```
42 -     x = 0:0.1:10;
43 -     y = x.^2;
44 -     z = sin(x^2).*cos(y^3);
45 -     plot(x,z)
46
```

Command Window

```
Error using  ^
Inputs must be a scalar and a square matrix.
To compute elementwise POWER, use POWER (.^)
instead.

Error in chapter2 (line 44)
z = sin(x^2).*cos(y^3);
```

Vectorized Functions:



Array Addressing

Now that you know how to create arrays, you need to be able to use specific elements, or rows, or columns in calculations. This is called **Array Addressing.** The **Colon** operator allows us to select individual elements, rows, columns, or parts of arrays.

Editor - C:\Users\scott\Desktop\chapter2.m

chapter2.m ×

```
K = [1 2; 3 4; 5 6];
L = [7 8 9; 10 11 12];
M = K*L

a = M(2,3)
b = M(:,3)
c = M(3,:)
d = M(2:3,1:2)
```

Command Window

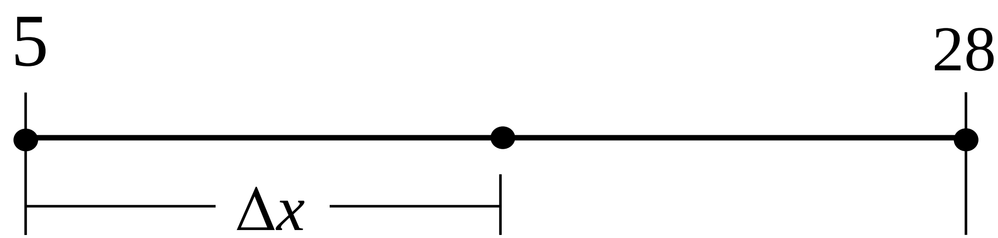
```
M =

    27    30    33
    61    68    75
    95   106   117


a =

    75


b =

    33
    75
   117


c =

    95   106   117


d =

    61    68
    95   106
```

Problem 2.1:

a. Use two methods to create the vector *x* having 100 regularly spaced values starting at 5 and ending at 28.

b. Use two methods to create the vector *x* having a regular spacing of 0.2 starting at 2 and ending at 14.

Start by looking at a simpler scenario, where three regularly spaced values are desired ($n = 3$):



$$\Delta x = \frac{(28 - 5)}{(3 - 1)} = 11.5$$

$$5 + 11.5 = 16.5$$

$$16.5 + 11.5 = 28$$

In General,

$$\Delta x = \frac{(x_{\max} - x_{\min})}{(n - 1)}$$

Command Window

```
xmin =

     5


xmax =

    28


N =

     3


deltax =

   11.5000


x =

    5.0000   16.5000   28.0000
```

Editor - C:\Users\scott\Desktop\Problem2_1.m

chapter2.m × | Problem2_1.m ×

```
clc
clear
xmin = 5
xmax = 28
N = 3
deltax = (xmax - xmin)/(N-1)
x = 5:deltax:28
```

Editor - C:\Users\scott\Desktop\Problem2_1.m

chapter2.m × | Problem2_1.m ×

```
clc
clear
xmin = 5
xmax = 28
N = 3
deltax = (xmax - xmin)/(N-1)
x = 5:deltax:28
x2 = linspace(5,28,N)
```

Command Window

```
xmin =

     5


xmax =

    28


N =

     3


deltax =

   11.5000


x =

    5.0000   16.5000   28.0000


x2 =

    5.0000   16.5000   28.0000
```

Problem 2.1:

a. Use two methods to create the vector *x* having 100 regularly spaced values starting at 5 and ending at 28.

b. Use two methods to create the vector *x* having a regular spacing of 0.2 starting at 2 and ending at 14.

6. Type this matrix in MATLAB and use MATLAB to carry out the following instructions.

$$\mathbf{A} = \begin{bmatrix} 3 & 7 & -4 & 12 \\ -5 & 9 & 10 & 2 \\ 6 & 13 & 8 & 11 \\ 15 & 5 & 4 & 1 \end{bmatrix}$$

a. Create a 4×3 array **B** consisting of all elements in the second through fourth columns of **A**.

b. Create a 3×4 array **C** consisting of all elements in the second through fourth rows of **A**.

c. Create a 2×3 array **D** consisting of all elements in the first two rows and the last three columns of **A**.

8. Given the matrix

$$\mathbf{A} = \begin{bmatrix} 3 & 7 & -4 & 12 \\ -5 & 9 & 10 & 2 \\ 6 & 13 & 8 & 11 \\ 15 & 5 & 4 & 1 \end{bmatrix}$$

a. Find the maximum and minimum values in each column.
b. Find the maximum and minimum values in each row.

13.* Given the matrices

$$\mathbf{A} = \begin{bmatrix} 56 & 32 \\ 24 & -16 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 14 & -4 \\ 6 & -2 \end{bmatrix}$$

Use MATLAB to

a. Find the result of **A** times **B** using the array product.
b. Find the result of **A** divided by **B** using array right division.
c. Find **B** raised to the third power element by element.

34. The volume of a parallelepiped can be computed from $|\mathbf{A} \cdot (\mathbf{B} \times \mathbf{C})|$, where **A**, **B**, and **C** define three sides of the parallelepiped (see Figure P34). Compute the volume of a parallelepiped defined by $\mathbf{A} = 5\mathbf{i}$, $\mathbf{B} = 2\mathbf{i} + 4\mathbf{j}$, and $\mathbf{C} = 3\mathbf{i} - 2\mathbf{k}$.

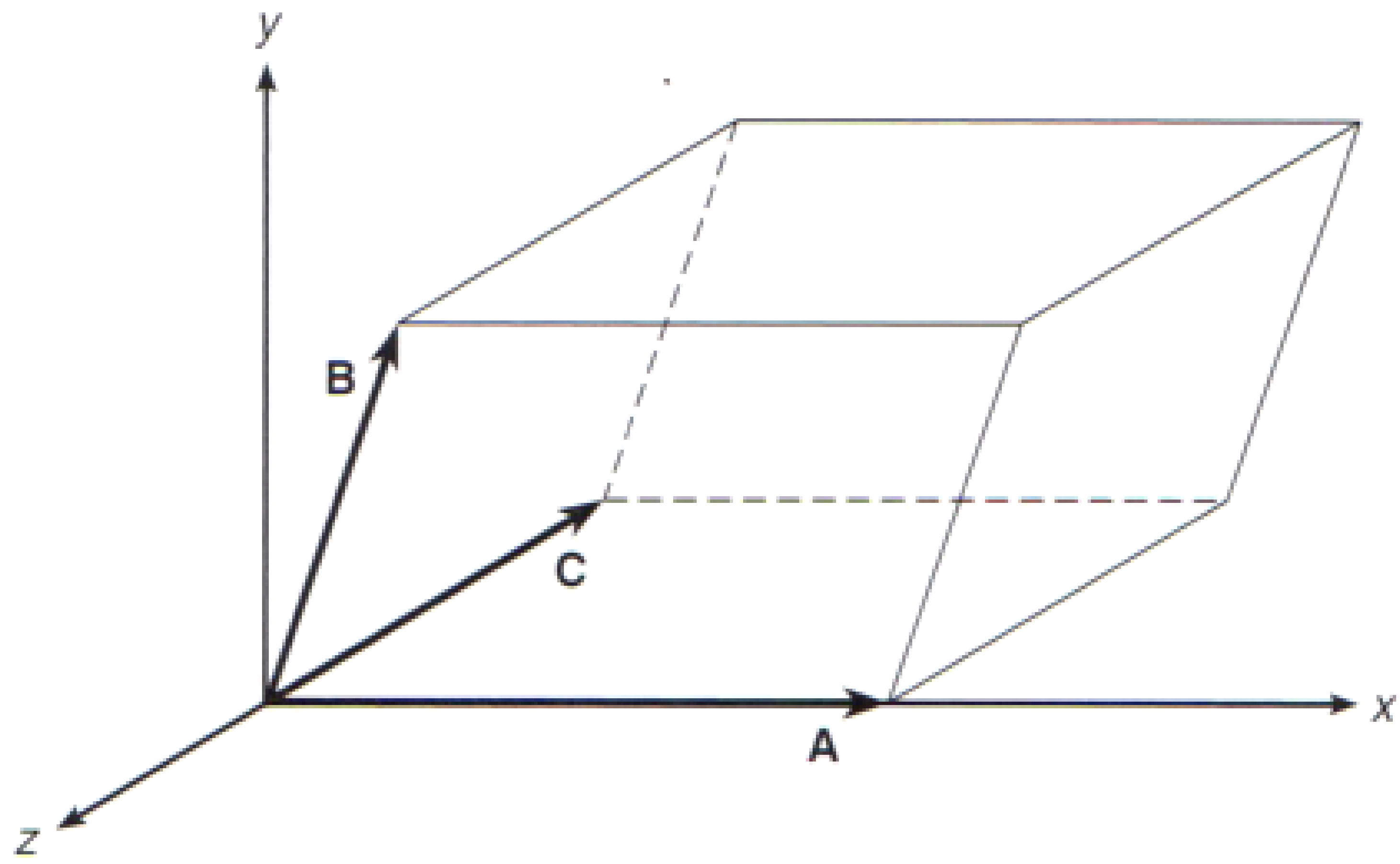


Figure P34