Probability and Statstics

Dr. Faisal Bukhari

Associate Professor

Department of Data Science

Faculty of Computing and Information Technology

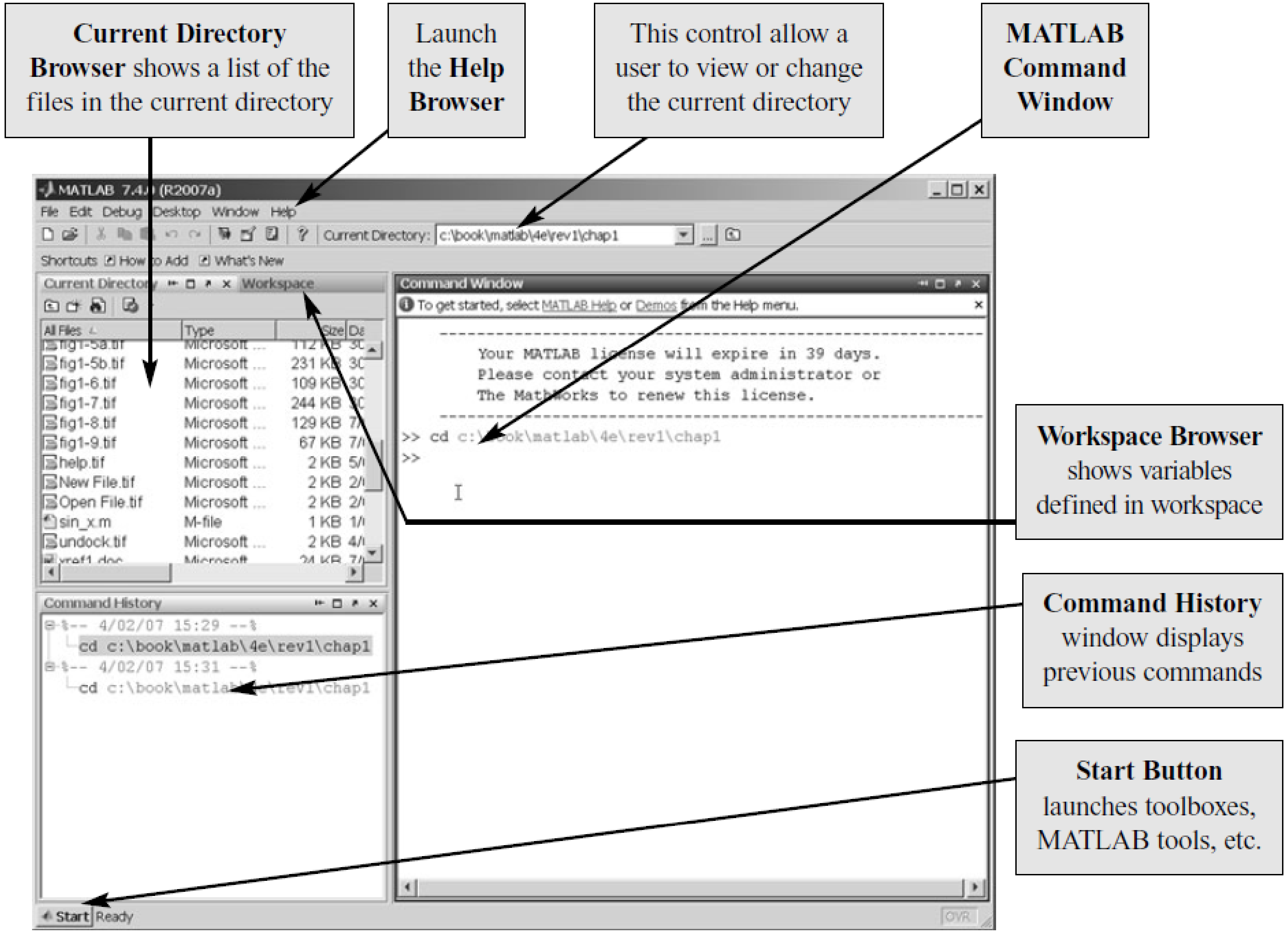
University of the Punjab

Textbooks

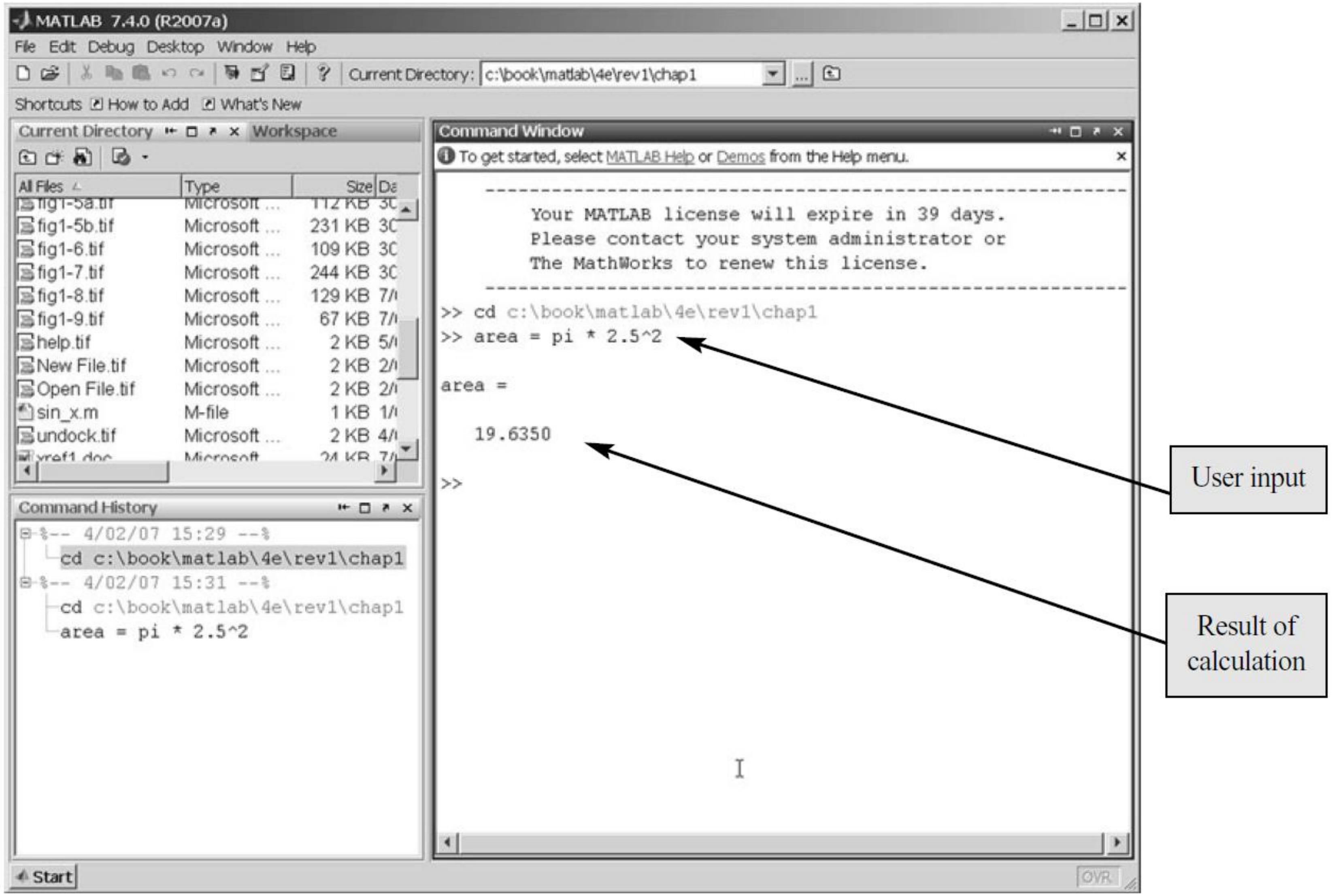
- **Probability & Statistics for Engineers & Scientists**, Ninth Edition, Ronald E. Walpole, Raymond H. Myer

Reference books

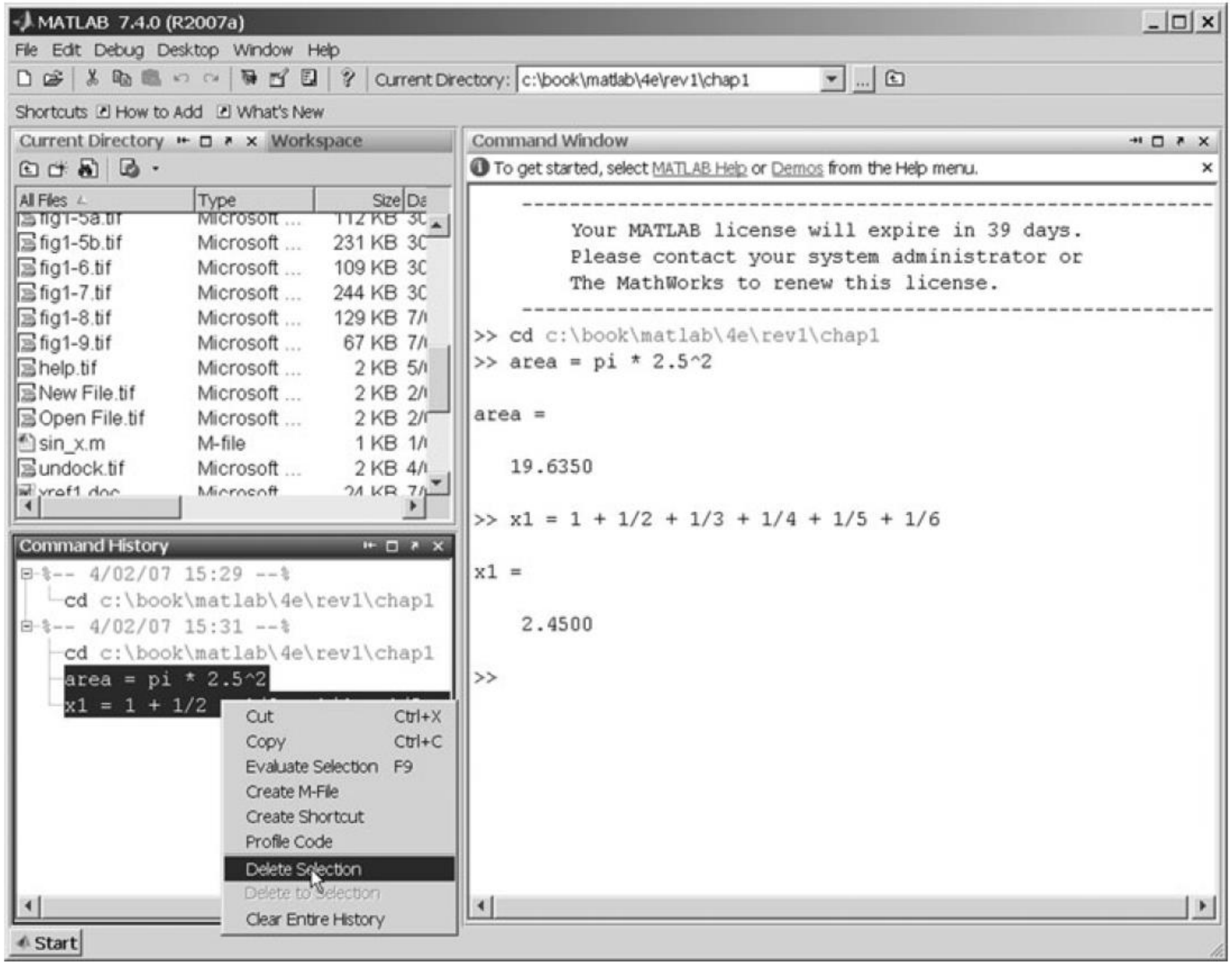
- ❑ **Probability Demystified**, Allan G. Bluman
- ❑ **Schaum's Outline of Probability and Statistics**
- ❑**MATLAB Primer**, Seventh Edition
- ❑**MATLAB Demystified** by McMahon, David



The default MATLAB desktop. The exact appearance of the desktop may differ slightly on different types of computers.



The Command Window appears on the right side of the desktop. Users enter commands and see responses here.



The Command History Window, showing two commands being deleted.

» lookfor inverse

INVHILB Inverse Hilbert matrix.

ACOS Inverse cosine.

ACOSH Inverse hyperbolic cosine.

ACOT Inverse cotangent.

ACOTH Inverse hyperbolic cotangent.

ACSC Inverse cosecant.

ACSCH Inverse hyperbolic cosecant.

ASEC Inverse secant.

ASECH Inverse hyperbolic secant.

ASIN Inverse sine.

ASINH Inverse hyperbolic sine.

ATAN Inverse tangent.

ATAN2 Four quadrant inverse tangent.

ATANH Inverse hyperbolic tangent.

ERFINV Inverse error function.

INV Matrix inverse.

PINV Pseudoinverse.

IFFT Inverse discrete Fourier transform.

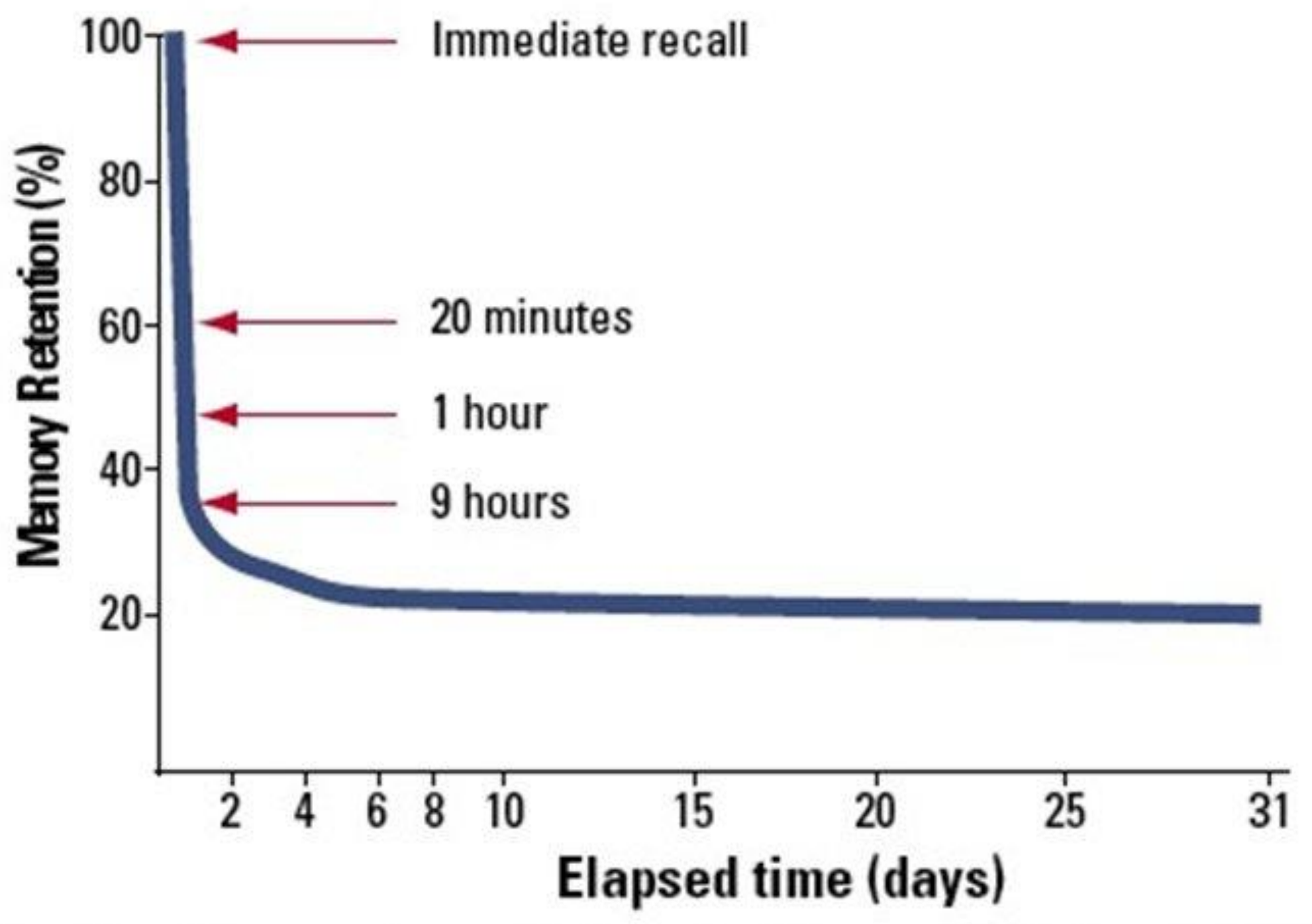
IFFT2 Two-dimensional inverse discrete Fourier transform.

IFFTN N-dimensional inverse discrete Fourier transform.

IPERMUTE Inverse permute array dimensions.

From this list, we can see that the function being sought is named inv.

Forgetting curve



Inverse of matrix using Matlab

```
A = [1 2 3 4; 600 700 8002 2; 3 4 8 7; 56 3737 9 22]
B = inv(A)
B =

  -1.6816   -0.0003    0.9612   -0.0001
   0.0216    0.0000   -0.0132    0.0003
   0.1241    0.0001   -0.0709   -0.0000
   0.5666   -0.0000   -0.1806   -0.0001
```

Solution of system of equations

Solve the system of equation:

$x + y = 10$

$x - y = 2$

Matlab code:

```
A=[1 1; 1 -1]

B=[10 ; 2]

X=A\B

X =

   6
   4
```

Predicted team performances for T20 World Cup 2016

```
x = [0.3, 0.2, 0.2, 0.04, 0.02, 0.02, 0.21, 0.01];

labels = {'Pakistan', 'India', 'Australia', 'South Africa',  'Sri Lanka', 'New Zealand', 'West Indies', 'Bangladesh'};

figure

pie(x, labels)
```

MATLAB (Matrix Laboratory) [1]

- MATLAB **(matrix laboratory)** is a multi-paradigm numerical computing environment and fourth-generation programming language.

- Developed by MathWorks, MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and **interfacing** with programs written in other languages, including **C, C++, Java**, and **Fortran**.

Reference: http://en.wikipedia.org/wiki/MATLAB

MATLAB (Matrix Laboratory) [2]

- Although MATLAB is intended primarily for numerical computing, an optional toolbox uses the MuPAD symbolic engine, allowing access to symbolic computing capabilities. An additional package, Simulink, adds graphical multi-domain simulation and Model-Based Design for dynamic and embedded systems.

- In 2004, MATLAB had around **one million users** across industry and academia[3]. MATLAB users come from various backgrounds of engineering, science, and economics. MATLAB is widely used in **academic** and **research institutions** as well as **industrial enterprises**.

Reference: http://en.wikipedia.org/wiki/MATLAB

Command Window and Basic Arithmetic [1]

The **Command Window** is found on the right-hand side of the MATLAB desktop. Commands are entered at the prompt with looks like two successive "greater than" signs:

>>

```
>> 433.12*15.7
    ans = 6.8000e+003
```

MATLAB spits out the answer to our query conveniently named **ans**. This is a variable or symbolic name that can be used to represent the value later.

```
>> x=5*6
     x = 30
```

Command Window and Basic Arithmetic [2]

Once a variable has been entered into the system, we can refer to it later. Suppose that we want to compute a new quantity that we'll call y, which is equal to x multiplied by 3.56. Then we type

```
>> y = x * 3.56
    y = 106.8000
```

Command Window and Basic Arithmetic [3]

- To write the multiplication ab, in MATLAB we type ab, in MATLAB we type **a * b**

- For division, the quantity **a ÷ b** is typed as **a / b.** This type of division is referred to as **right division**.

- MATLAB also allows another way to enter division, called **left division.** We can enter the quantity b by a typing the slash mark used for division in the opposite way, that is, we use a back slash instead of a forward slash **a \ b**

Command Window and Basic Arithmetic [4]

Exponentiation **a to the power b** is entered in the following way **a ^ b**

Finally, **addition** and **subtraction** are entered in the usual way

a + b

a – b

Referencing individual entries

Individual matrix and vector entries can be referenced with indices inside parentheses. For example, A(2,3) denotes the entry in the second row, third column of matrix A.

A = [1 2 3 ; 4 5 6 ; -1 7 9]

A(2,3)

Create a column vector, x, with:

x = [3 2 1]’

or equivalently:

x = [3 ; 2 ; 1]

Two-dimensional arrays:

Format: A = [row1 ; row2 ; ... ; rowN] an array with N rows.

- Entry of arrays begins with a left square bracket, **[** , and ends with a right square bracket, **]**.

- Within the brackets rows of numbers are separated by semicolons, **;** , each row must have the same number of elements and elements of the **rows are separated** by **spaces** or **commas**.

Example 1

A=[2 3 4; -5 4 2; 4 -3 5]

A =

2	**3**	**4**
-5	**4**	**2**
4	**-3**	**5**

Example 2

A2= [1 3 4

-5 54 2

4 -3 15

2 2 -9]

A2 =

1	**3**	**4**
-5	**54**	**2**
4	**-3**	**15**
2	**2**	**-9**

Display command

The display command: disp(variable).

- To **display *A*** , enter ***disp(A),*** to display ***A*** and ***A2*** together enter ***disp(A); disp(A2)***

Example 3

disp(A); disp(A2)

2	3	4
-5	4	2
4	-3	5

1	3	4
-5	54	2
4	-3	15
2	2	-9

Revision of array elements:

- **Revision of array elements: Assignment: change a single element.**

- Use row and column position numbers as an argument of the array variable name.

- Note: the use of the **colon (wild card)** in arrays, **:** , means to **use all elements of a specified row or column**.

- The notation, **A(:,:)** represents **all elements of array A** .

- To change the row 2 column 3 element of A from 2 to 17, type **A(2,3)=17**

Example 4

A(2,3) = 17

A =

$$\begin{matrix} 2 & 3 & 4 \\ -5 & 4 & 17 \\ 4 & -3 & 5 \end{matrix}$$

Recall

A =

$$\begin{matrix} 2 & 3 & 4 \\ -5 & 4 & 2 \\ 4 & -3 & 5 \end{matrix}$$

Example 5

To replace **row 3** of *A* by **[1 23 8],** type **A(3,:) = [1 23 8]**

A =

2	**3**	**4**
-5	**4**	**17**
1	**23**	**8**

Recall

A =

2	**3**	**4**
-5	**4**	**17**
4	**-3**	**5**

Example 6

To replace **column 2** of *A* by **[4 6 -14],** type **A(:,2)=[4 6 -14]**

A =

2	**4**	**4**
-5	**6**	**17**
1	**-14**	**8**

Recall

A =

2	**3**	**4**
-5	**4**	**17**
1	**23**	**8**

Operations in MATLAB

if we want to do element wise operation then we use '.' before symbols of operations.

Symbol	Operation	Meaning
+/-	addition/subtraction	elementwise addition or subtraction
*	multiplication	standard matrix multiplication
.*	multiplication	elementwise multiplication, e.g., [4 3].*[2 -2] = [8 -6]
/	division	B/A = (A'\B')', solution of X*A=B
. /	Division	. / Division elementwise division, e.g., [4 3] . / [2 -2] = [2 -1.5]
\	None	A\B, efficient solution of A*X=B
^	exponentiation	matrix exponentiation, A^3=A*A*A
. ^	exponentiation	elementwise exponentiation, e.g., [4 3] . ^ 3 = [64 27]

Augmenting arrays

Symbol	Action (meaning)
;	Begin new row of array, separate MATLAB commands on a line
:	Wildcard for all rows or all columns, separator for limits
,	Separator for elements or array parts

Rowwise augmentation:

[B ; C], the new augmented array has *B* on top and *C* on bottom, if *B* and *C* **have the same number of columns**.

If *B* = [1 2], *C* = [3 4], then

B=[1 2];C=[3 4];D =[B;C]

D =

1	**2**
3	**4**

Columnwise Augmentation

[B C], the new array has B and C side by side, if B and C have the **same number of rows**. If B = [1 2], C = [3 4],

D= [B C]

D =

1 2 3 4

Hilbert matrix

- **hilb** Hilbert matrix.
- **hilb(N)** is the N by N matrix with elements **1/(i+j-1).**

Format command

- **format Set output format.**
- format with no inputs sets the output format to the default appropriate for the class of the variable. For float variables, the default is format **SHORT**.

format SHORT	Scaled fixed point format with 5 digits.
format LONG	Scaled fixed point format with 15 digits for double and 7 digits for single.
format SHORTE	Floating point format with 5 digits.
format LONGE	Floating point format with 15 digits for double and 7 digits for single.
format SHORTG	Best of fixed or floating point format with 5 digits.
format LONGG	Best of fixed or floating point format with 15 digits for double and 7 digits for single.
format SHORTENG	Engineering format that has at least 5 digits and a power that is a multiple of three
format LONGENG	Engineering format that has exactly 16 significant digits and a power that is a multiple of three.

format RAT Approximation by ratio of small integers. Numbers with a large numerator or large denominator are replaced by *.

```
format short, pi
ans =
   3.1416
format long, pi
ans =
  3.141592653589793
```

format rat; D = hilb(6) format rat, D = hilb(6) => this is also a valid syntax

```
D =

    Columns 1 through 3                Columns 4 through 6

    1        1/2      1/3              1/4      1/5      1/6
    1/2      1/3      1/4              1/5      1/6      1/7
    1/3      1/4      1/5              1/6      1/7      1/8
    1/4      1/5      1/6              1/7      1/8      1/9
    1/5      1/6      1/7              1/8      1/9      1/10
    1/6      1/7      1/8              1/9      1/10     1/11
```

The command ***E = D(2:4, 5:6)*** stores elements from rows 2 through **4** and in columns **5** through **6** of ***D*** into ***E***.

Note: anything entered after the % symbol is a comment and is ignored.

D = hilb(6);format rat; E = D(2:4,5:6)

E =

1/6	**1/7**
1/7	**1/8**
1/8	**1/9**

F = D(:, 4:6)

%stores elements from all rows and in columns 4 through 6 of *D* into *F*.

F =

1/4	**1/5**	**1/6**
1/5	**1/6**	**1/7**
1/6	**1/7**	**1/8**
1/7	**1/8**	**1/9**
1/8	**1/9**	**1/10**
1/9	**1/10**	**1/11**

G = D(2:4,:)

%stores the elements in rows 2 through 4, all columns, of *D* into *G*.

G =

Columns 1 through 3

1/2	**1/3**	**1/4**
1/3	**1/4**	**1/5**
1/4	**1/5**	**1/6**

Columns 4 through 6

1/5	**1/6**	**1/7**
1/6	**1/7**	**1/8**
1/7	**1/8**	**1/9**

PICKING OFF PARTS OF ARRAYS: HOW TO SELECT SUB-ARRAYS

For the **array *D***, the command **D(*row i : row j , column k : column l*)**, picks off subarrays of array *D*.