ELG3125 Signal and System Analysis

Introduction to MATLab

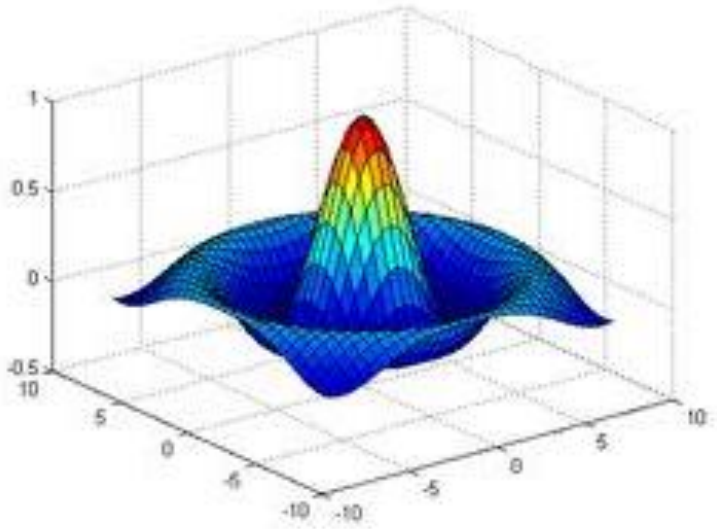
TA: Hiva Shahoei

(hshahoei@uottawa.ca)

Fall, 2011

Outline

1. MATLab Overview
2. Desktop and Interfaces
3. Basics
4. Help



MATLab Overview

MATLab (MATrix LABoratory) is a high-performance language for technical computing. Typical uses include:

- Math and computation,
- Algorithm development
- Data acquisition,
- Modeling, simulation, and prototyping
- Data analysis, exploration, and visualization
- Scientific and engineering graphics

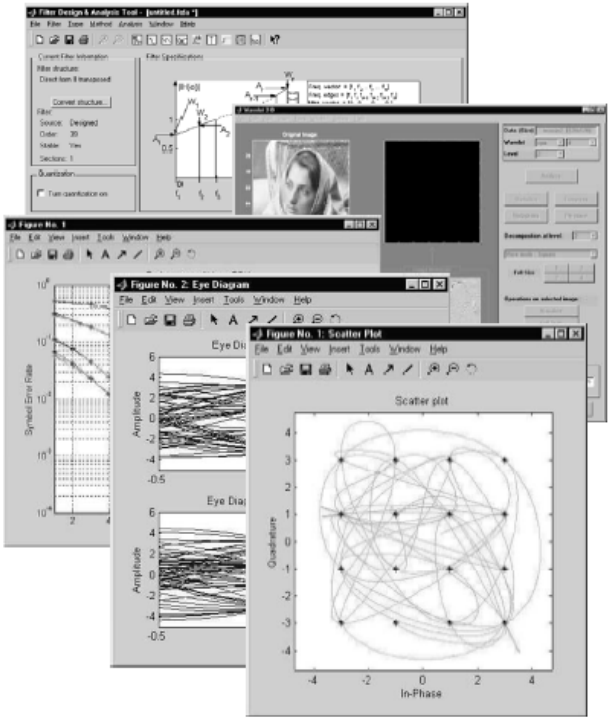
MATLab Overview

A high-level matrix/array language with

- control flow statements,
- functions,
- data structures,
- input/output,
- and object-oriented programming features.

MATLab Overview

Toolboxes

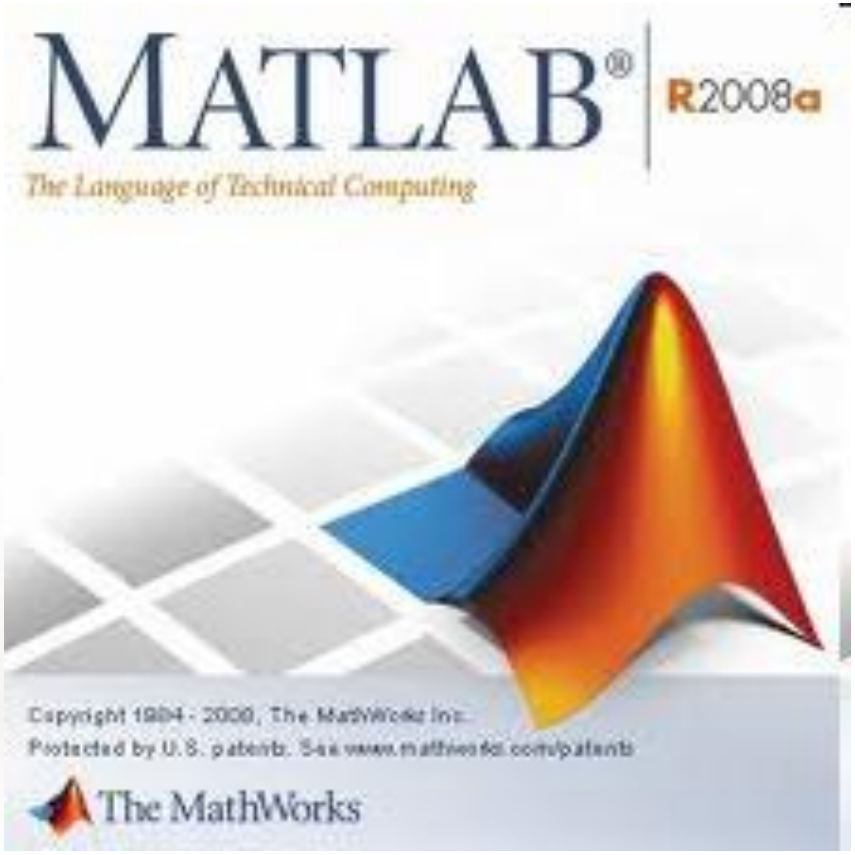


- Signal Processing
- Communications
- Filter Design
- Wavelet Analysis
- Statistics
- Optimization
- Image Processing
- Others…

Start to run MATLab……

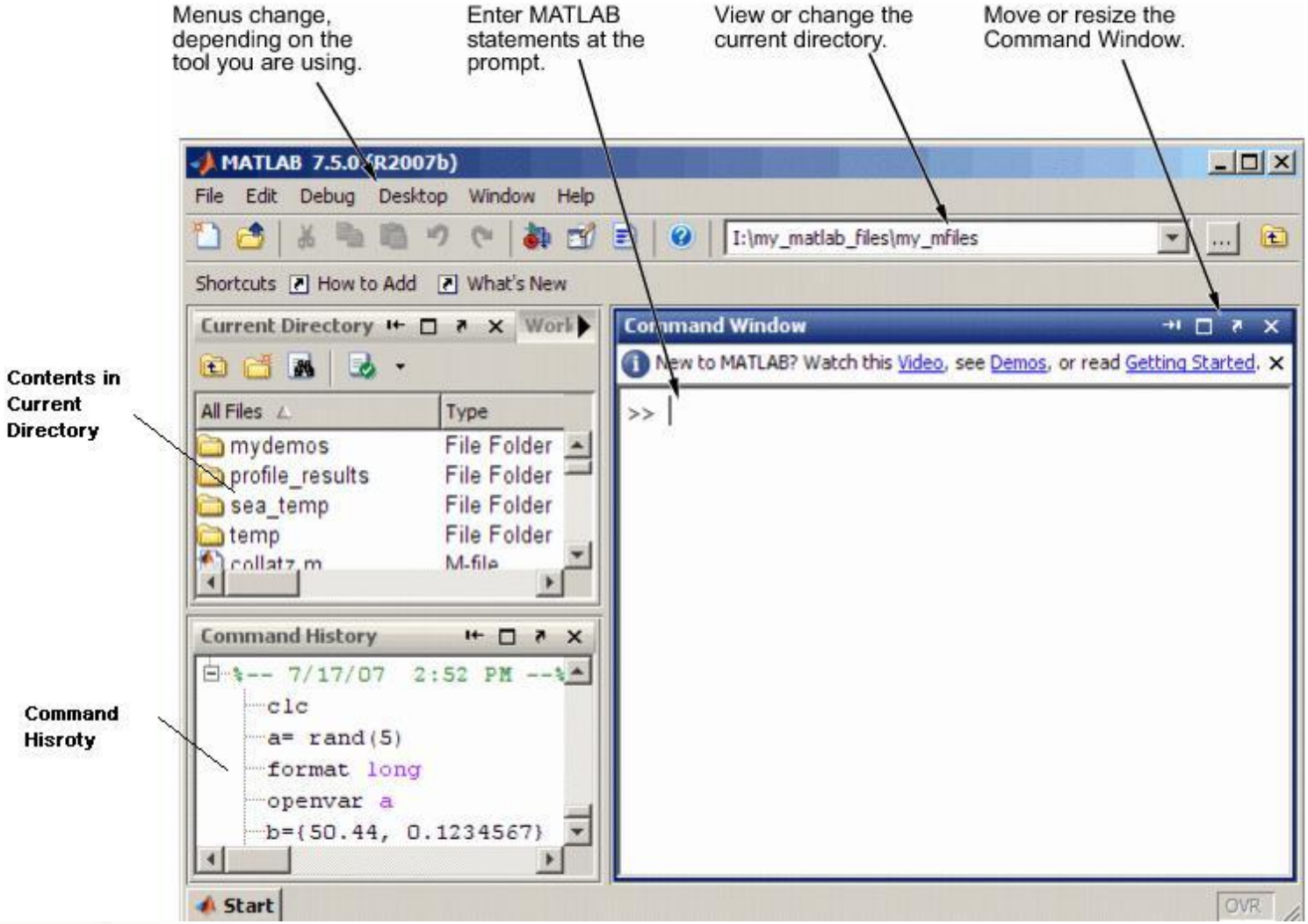
- Starting MATLab and a screen similar to the right one appears, followed by the desktop interface.

.



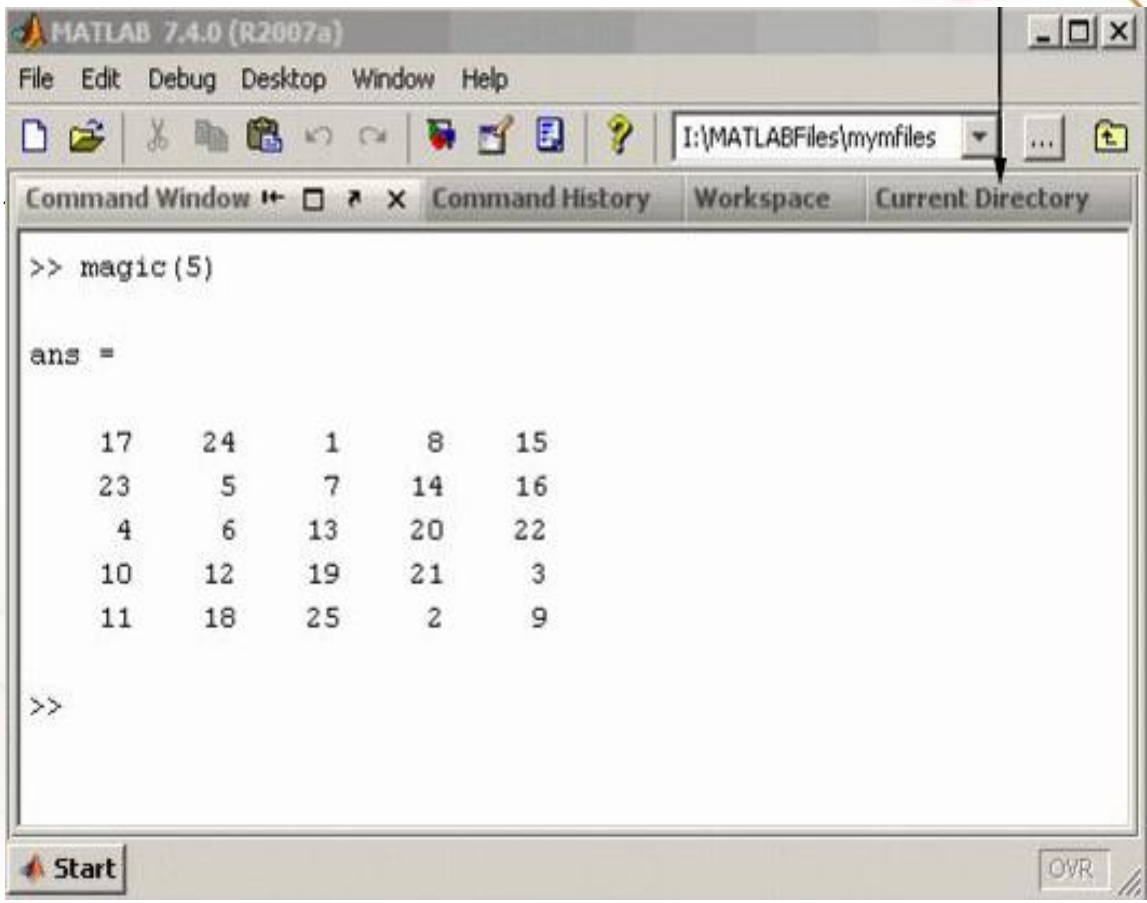
Desktop and Interfaces (Default)

- Command Window
- Command History
- Current Directory
- Workspace



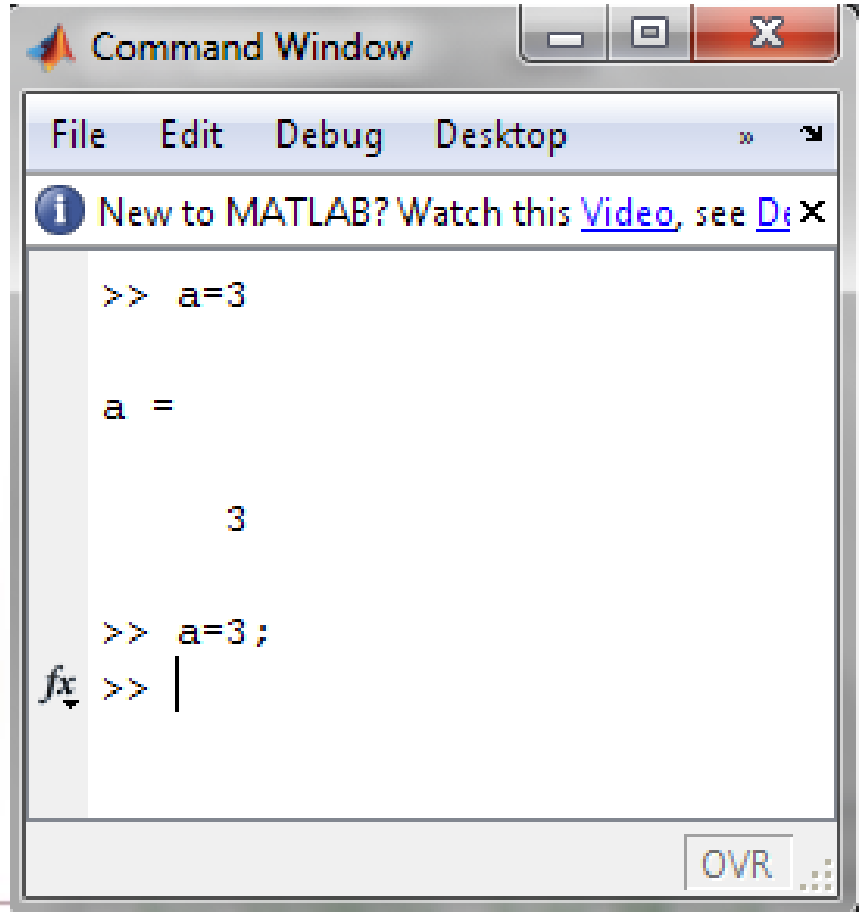
Desktop and Interfaces

- Rearrange the desktop, for example one layout like the left one. It gives more space for each window.



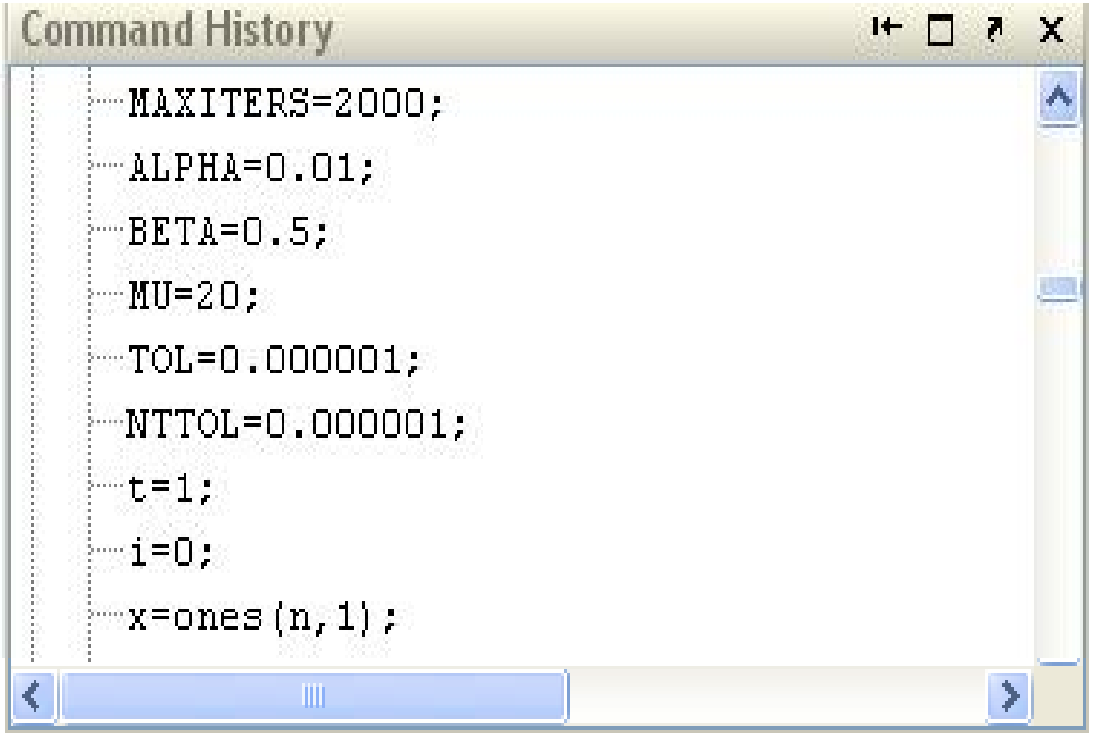
Command Window

- Prompt >> appears and Matlab command can be entered.
- Press <ENTER> to execute the Matlab command and next prompt >> appears.
- If a command is terminated with a semicolon (;), no results will be displayed.
- Otherwise results will be displayed before next prompt appears.



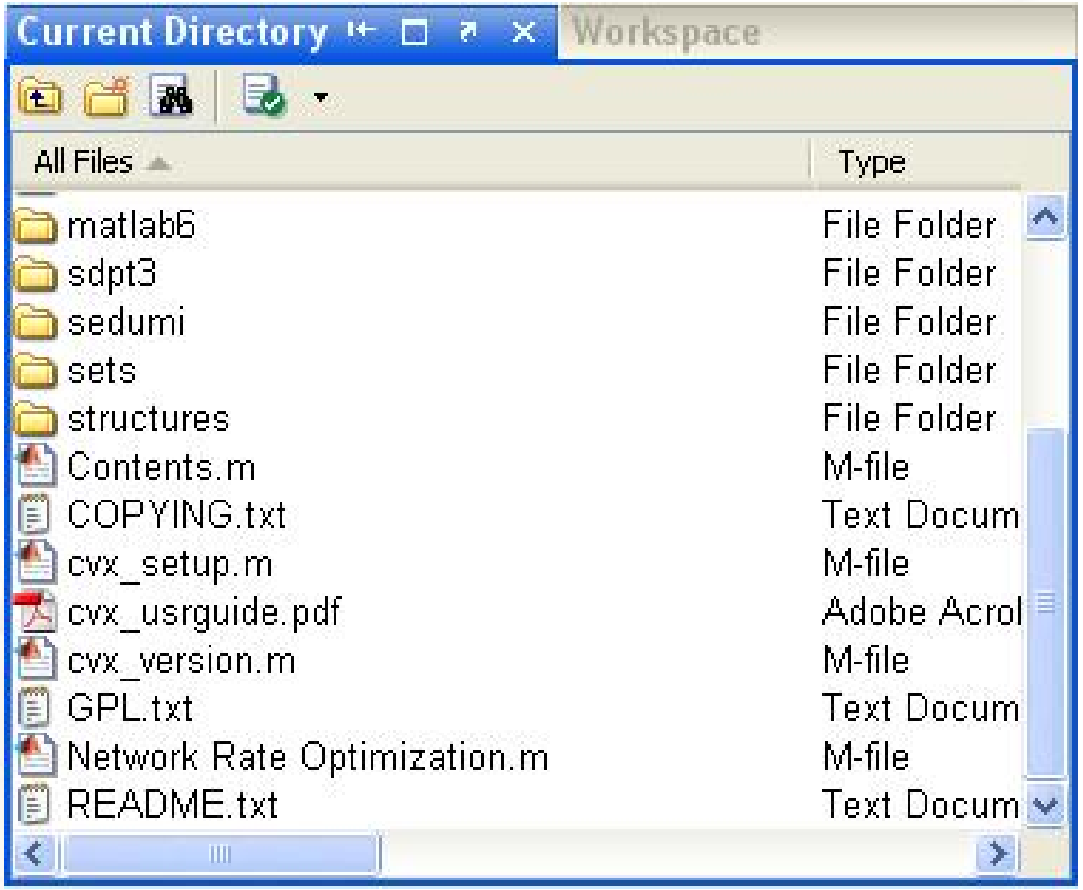
Command History

- Command History store the commands entered previously.
- Provide an easy way to repeat a command later on.



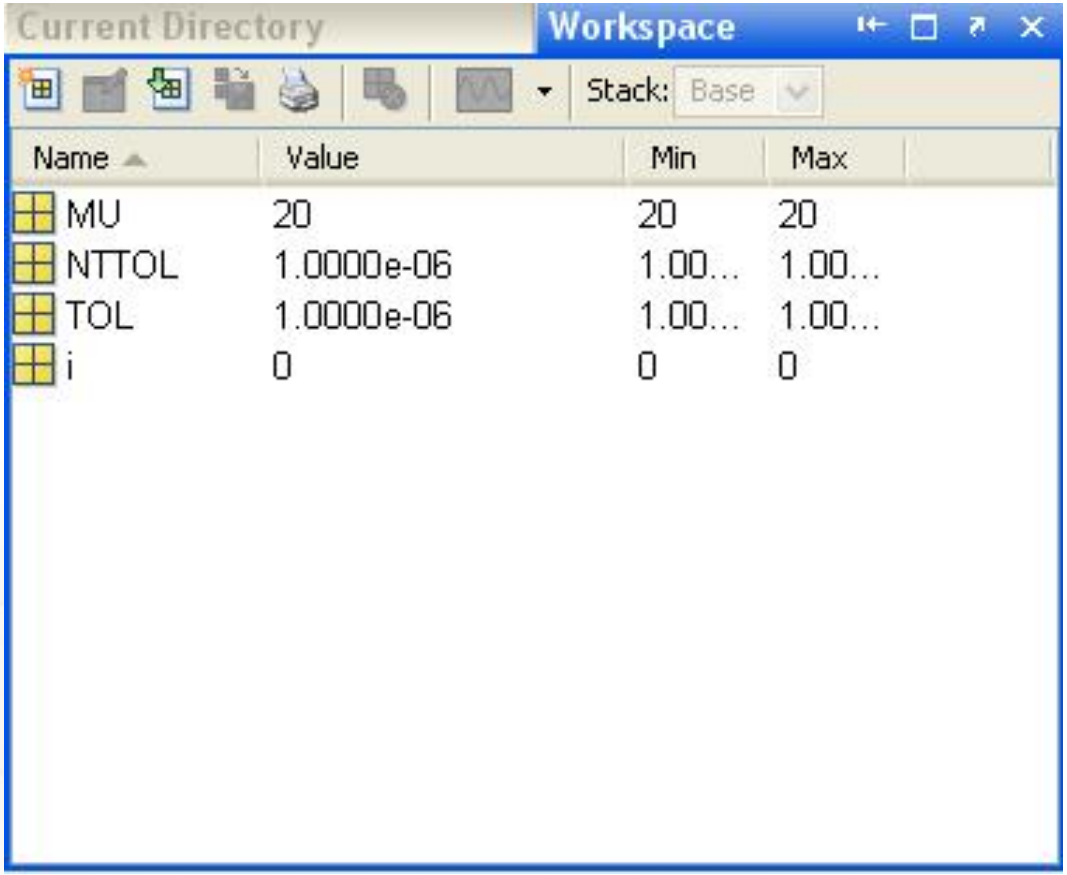
Current Directory

- Show all the folders and files in the current directory



Workspace

- The Workspace browser shows the name of each variable, its value, and the Min and Max calculations
- The MATLAB workspace consists of the set of variables (named arrays) built up during a MATLAB session



Matlab Symbols

- >> Command prompt
- … Continue statement in next line
- **,** Separate statements and data, e.g.,

  A = [5.92, 8.13, 3.53]
- **%** Start comment which ends at the end of line, e.g.,

  % The purpose of this routine is to compute

  % the value of ...
- **;** Suppress output or used as row separator in a matrix
- **:** Specify a range and **g**enerates a sequence of numbers that you can use in creating or indexing into arrays. For example, N = 6:17.

MATLab Variables

Variable name

- CASE SENSITIVE. e.g., x and X are two different variables
- up to 63 characters (as of MATLAB 6.5 and newer)
- must start with a letter and can be followed by letters, digits, or underscores. e.g.,x3_2, but 2_x3 is not correct.

MATLab Variables

- Variables are stored in Matlab in the form of matrices

  which are generally of size MxN.
- Elements of matrix can be real or complex numbers.
- A scalar is a 1x1 matrix.
- A row vector is a 1xN matrix.
- A column vector is a Mx1 matrix.

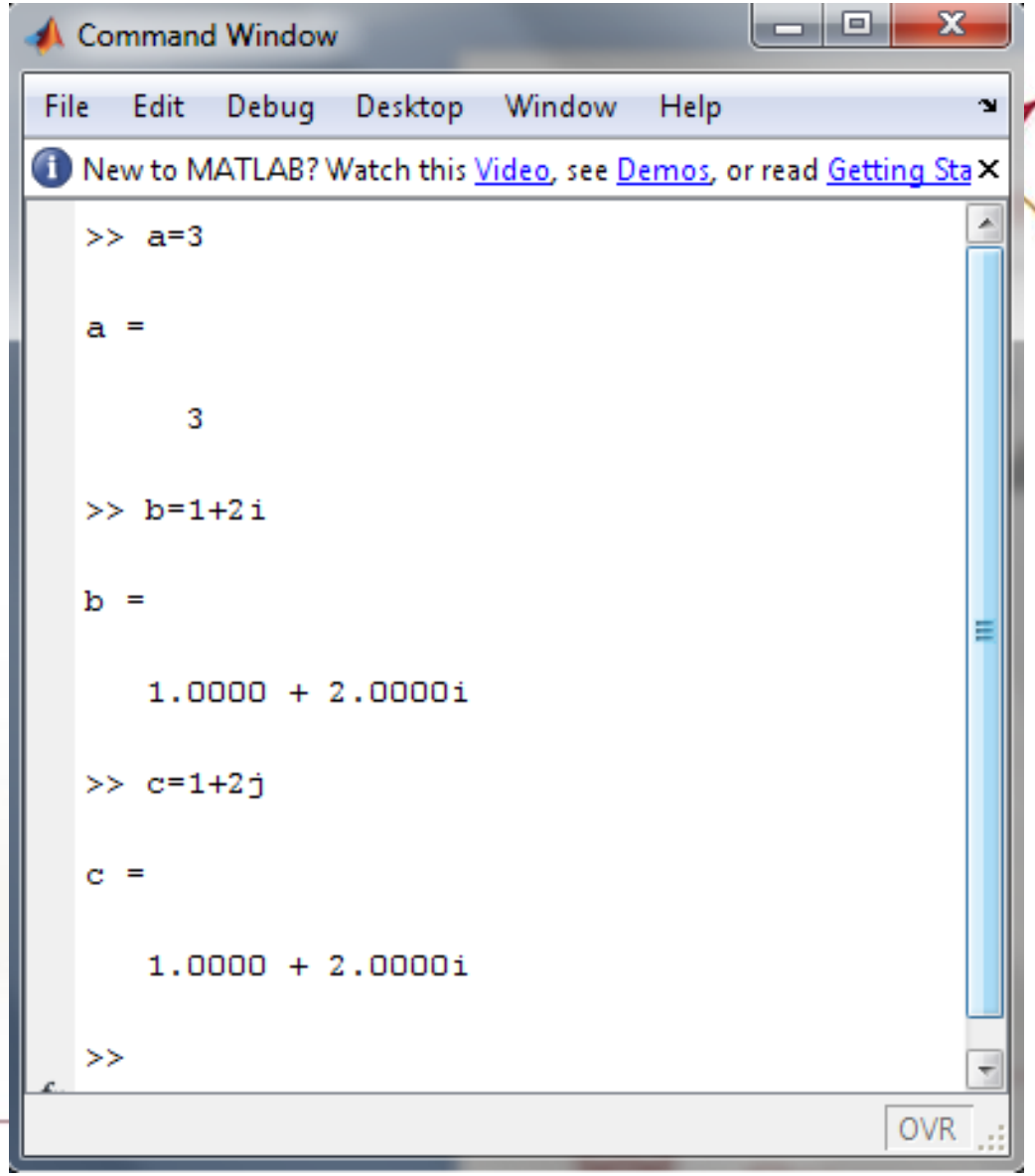
MATLab Variables

Special variables

- **ans** **Default variable name for results**
- **pi** **Value of π**
- **eps** **Smallest incremental number**
- **inf** **Infinity**
- **NaN** **Not a number e.g. 0/0**
- **i and j** **i = j = square root of -1**
- **realmin** **The smallest usable positive real number**
- **realmax** **The largest usable positive real number**

Scalars examples

- Real numbers
- Complex numbers



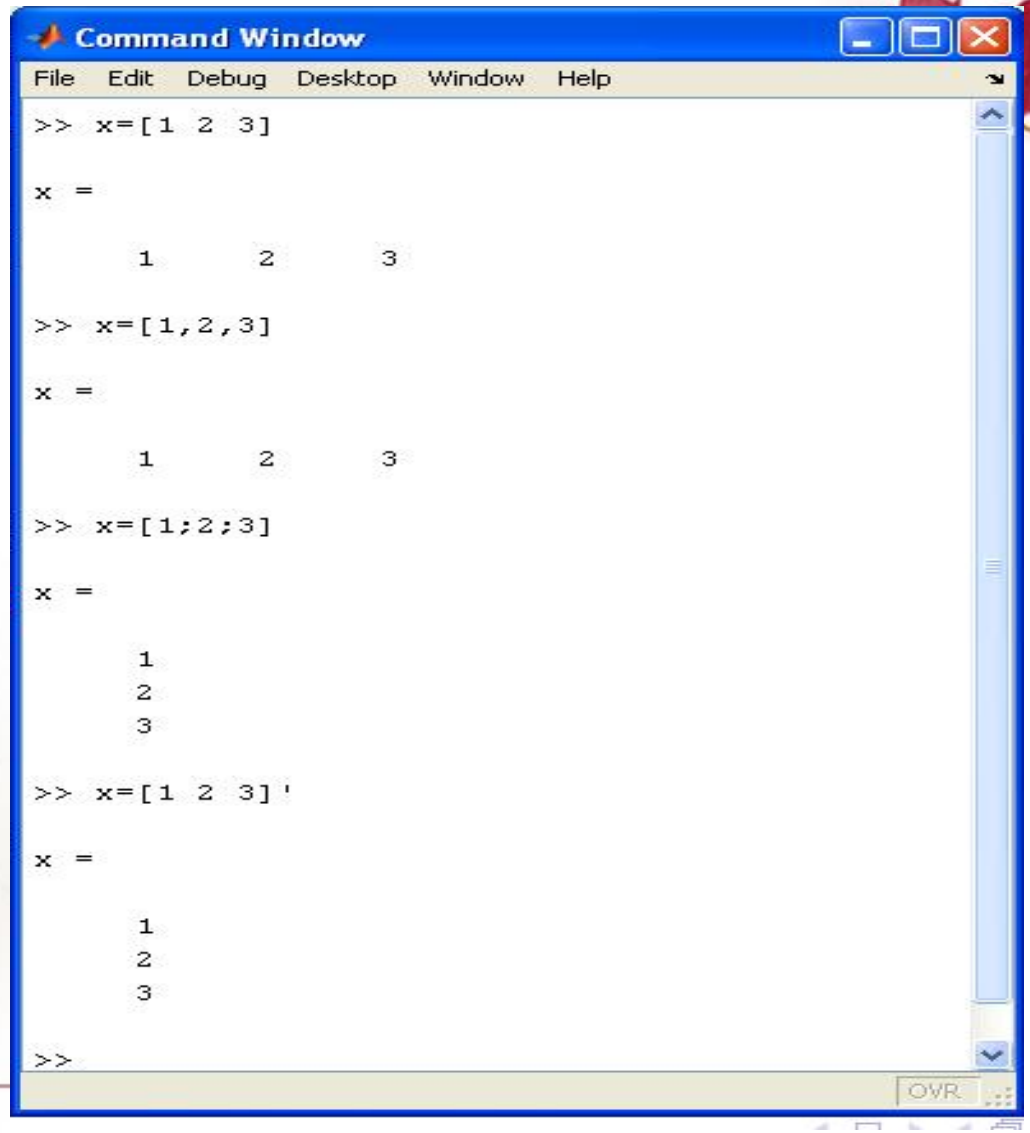
Vectors examples

- Row vector

  x=[1 2 3]

- Column vector

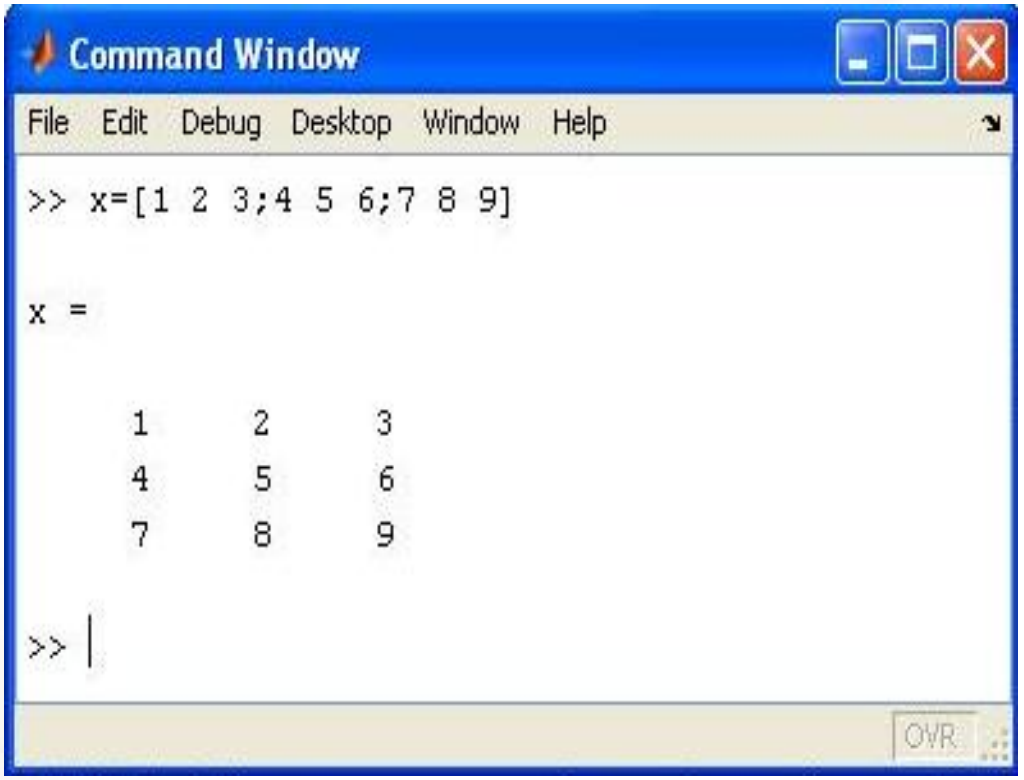
  x=[1;2;3]



Matrix example

x=[1 2 3;4 5 6;7 8 9]

- Matrix is input row by row.
- Row elements are separated by space or comma
- Different rows are separated by semicolon.



Zero matrix

- >> x=zeros(3)
- x =

0	0	0
0	0	0
0	0	0

- >> x=zeros(2,3)
- x =

0	0	0
0	0	0

One Matrix

- >> x=ones(3)

x =

1	1	1
1	1	1
1	1	1

>> x=ones(4,3)

x =

1	1	1
1	1	1
1	1	1
1	1	1

Identity Matrix (Unit Matrix)

```
>> x=eye(3)

x =

     1     0     0
     0     1     0
     0     0     1
```

```
>> y=eye(5)

y =

     1     0     0     0     0
     0     1     0     0     0
     0     0     1     0     0
     0     0     0     1     0
     0     0     0     0     1
```

Matrix Operation

```
>> A=[10 20 30;11 21 31]

A =

   10    20    30
   11    21    31

>> a=A(1,2)

a =

   20

>> a1=A(1,:)

a1 =

   10    20    30

>> a2=A(:,2)

a2 =

   20
   21

>> A1=A(1:2,1:2)

A1 =

   10    20
   11    21
```

Arithmetic Operator

Matrix Operation:

+ Addition

- Subtraction

* Multiplication

/ Right division

\ Left division

^ Power

’ Conjugate transpose

```
A=[10 20 30;11 21 31]
B=[1 1; 1 1; 1 1]

>> A*B

ans =

    60    60
    63    63
```

Arithmetic Operator

Element-by-element operations.

. * Multiplication

. / Right division

. \ Left division

. ^ Power

. ’ Transpose (unconjugated)

```
A=[10 20 30;11 21 31]
C=[1 1 1;1 1 1]

>> A.*C

ans =

    10    20    30
    11    21    31
```

Relational operators

- == equal
- ~= not equal
- < less than
- <= less than or equal
- > greater than
- >= greater than or equal

Flow control

- if-elseif-else statements
- switch-case statements
- for loops
- while loops

Basic Math Functions

• abs(x)	absolute value
• exp(x)	exponential
• sin(x),cos(x)	sine, cosine
• log(x),log10(x)	natural logarithm, common logarithm
• sqrt(x)	square root
• sign(x)	signum
• round(x),fix(x)	round towards nearest integer, round towards zero
• floor(x),ceil(x)	round towards negative infinity, round towards plus infinity
• size(x),length(x)	size of array, length of vector

Two types of M-Files

M-files are used to execute a series of commands.

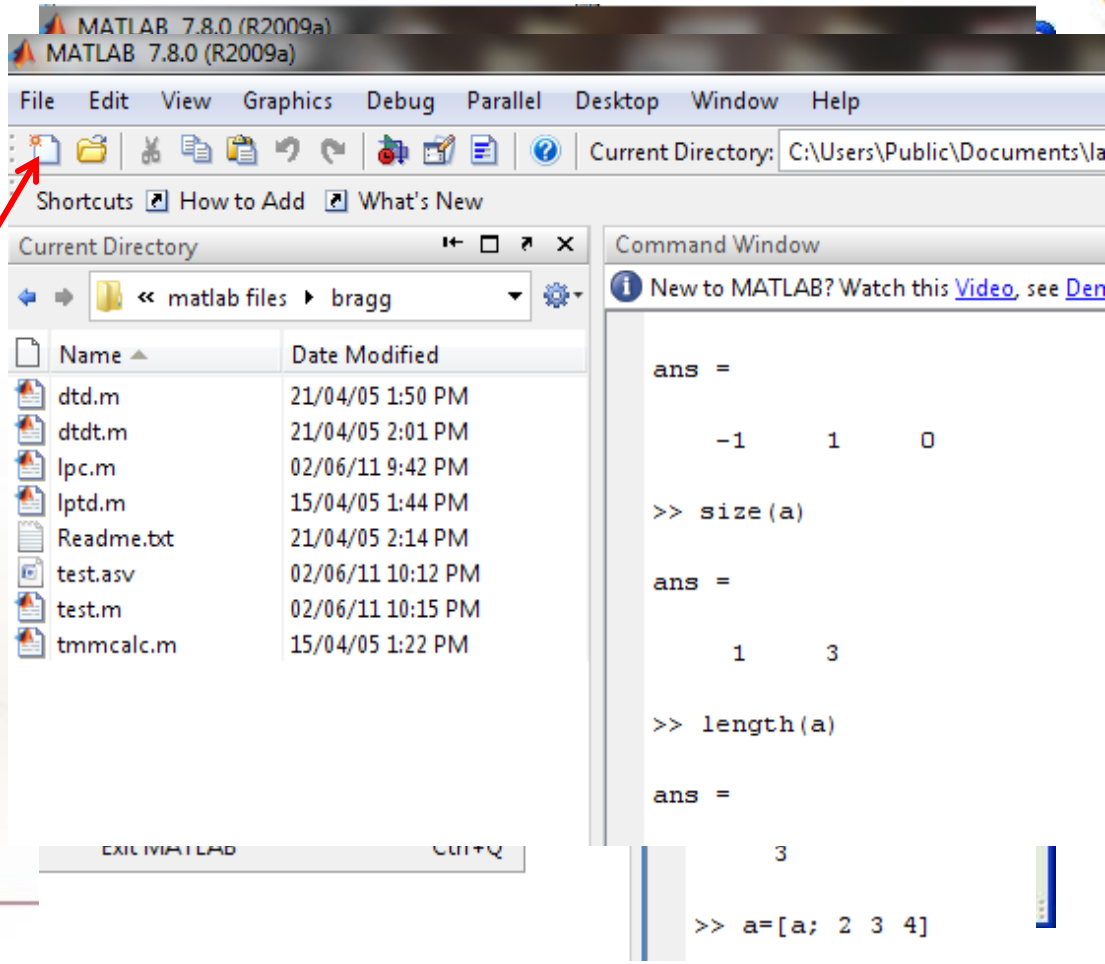
- Script: Running with No Input Arguments;
- Function: Running with Input Arguments;

✓Both have a .m extension.

✓Created in Editor Window.

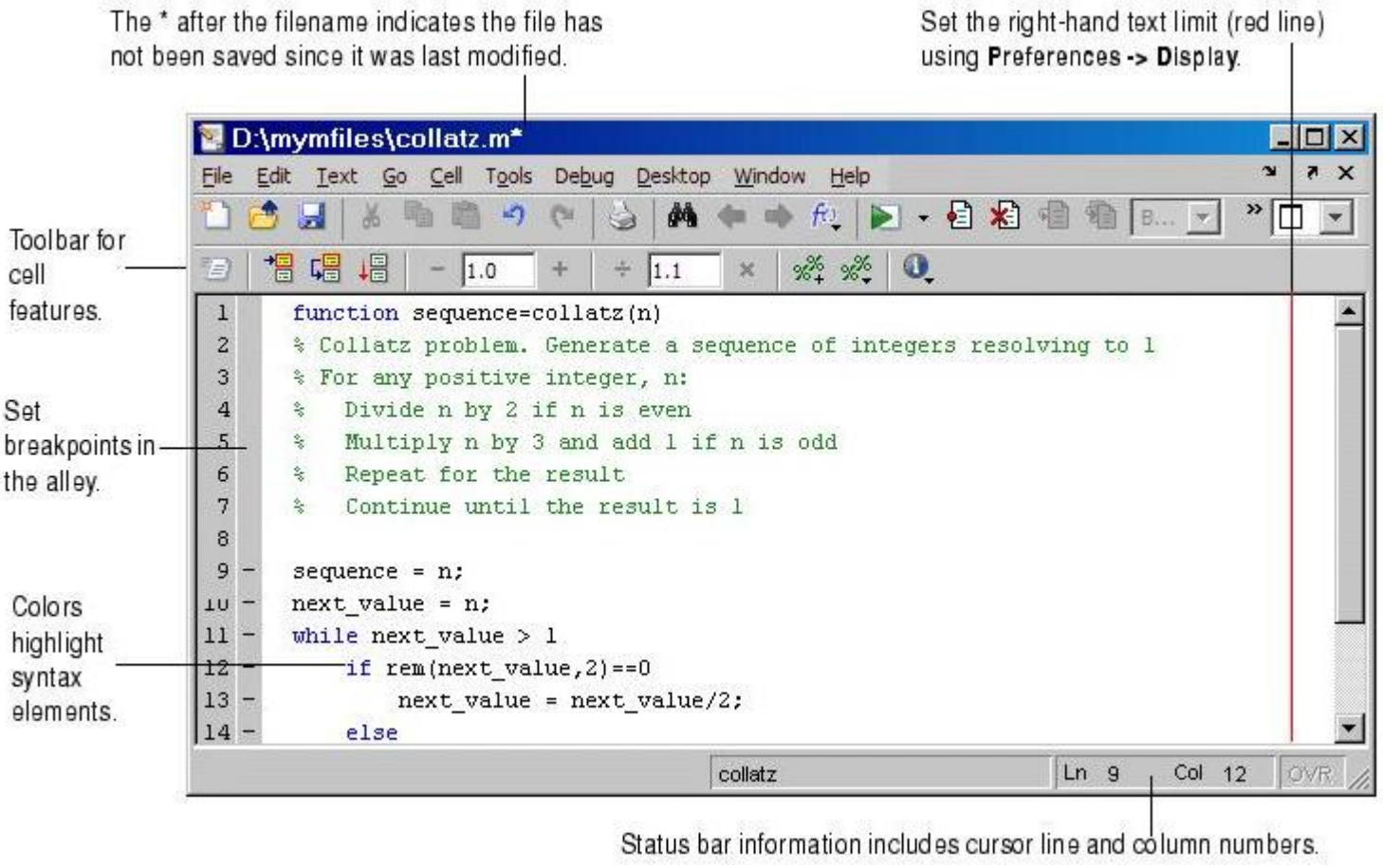
Creating a New Script M-File

- click the New M-file button on the MATLAB desktop toolbar;
- or select File > New > M-File from the MATLAB desktop.



Open an Existing M-File

- File>Open



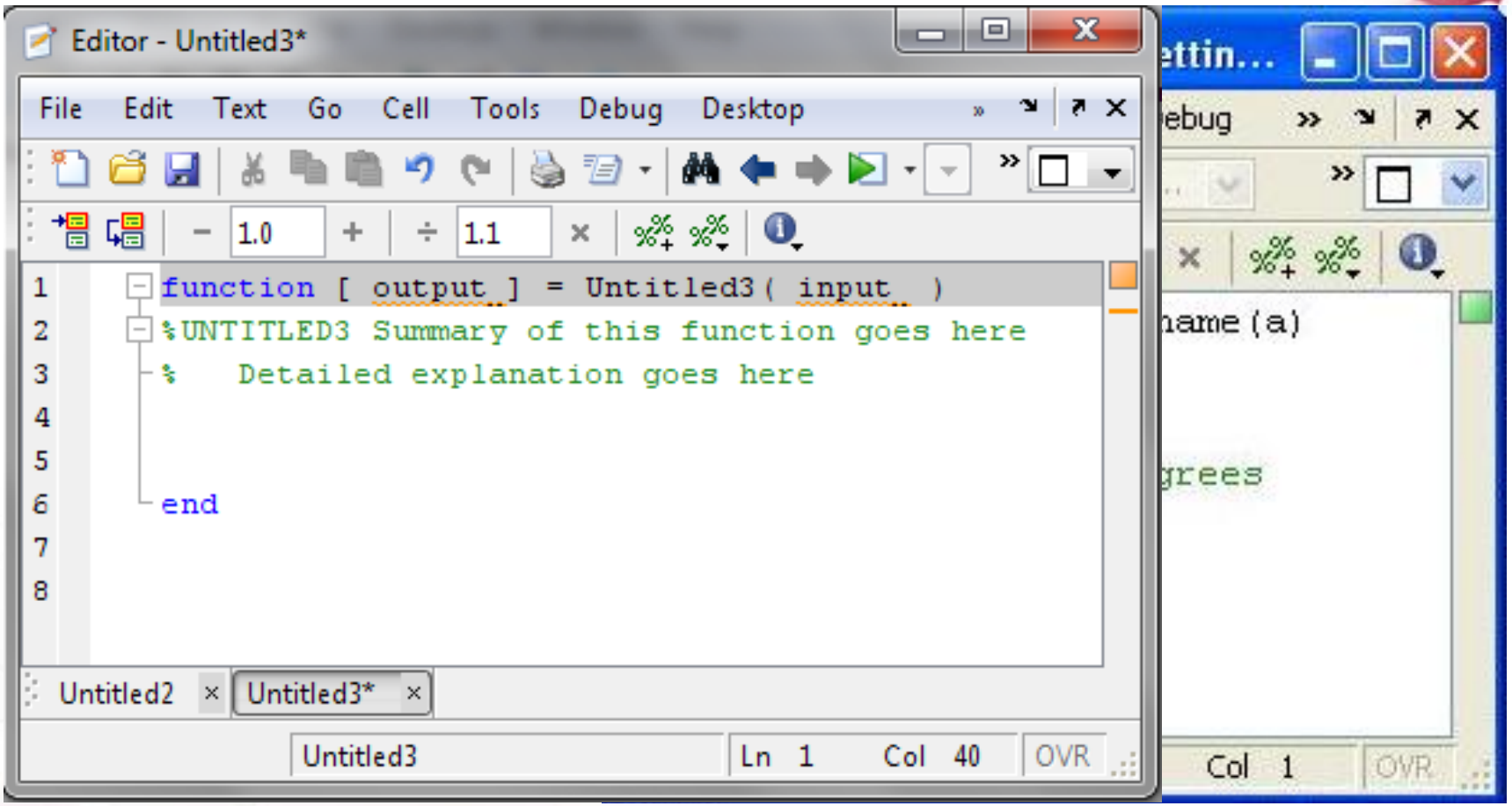
Example of a Script M-File

```
a=pi/3;    %60 degrees
x=sin(a);
b=pi/4;    %90 degrees
y=cos(b);
c=x*y
```

```
c =

    0.6124
```

Example of a function M-File



MATLab Graphing

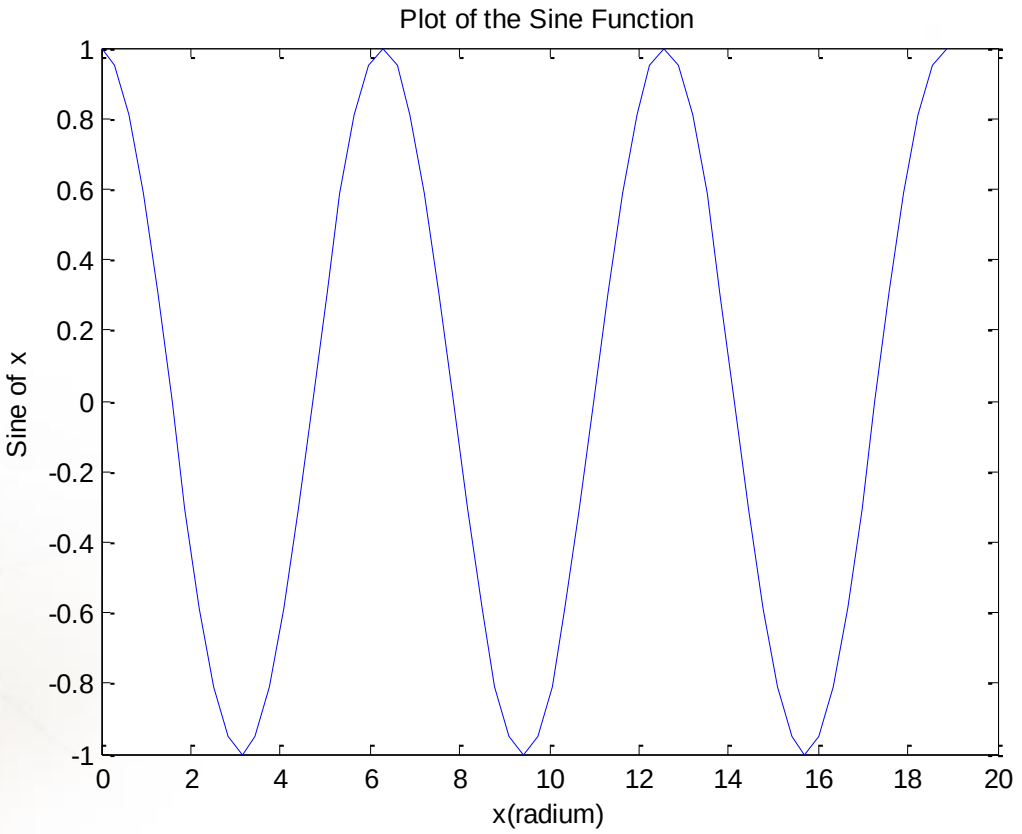
- plot Plot in linear coordinates as a continuous function
- stem Plot in linear coordinates as discrete samples
- loglog Logarithmic x and y axes
- semilogx Linear y and logarithmic x axes
- semilogy Linear x and logarithmic y axes
- xlabel Label x-axis
- ylabel Label y-axis
- title Put a title on the plot

MATLab Graphing (cont.)

- grid Add a grid to the plot
- text Allow placing text at specified coordinates of the plot
- axis Control axis scaling and appearance
- figure Create a figure for plotting
- hold Hold current graph
- subplot Divide figure window and select figure position

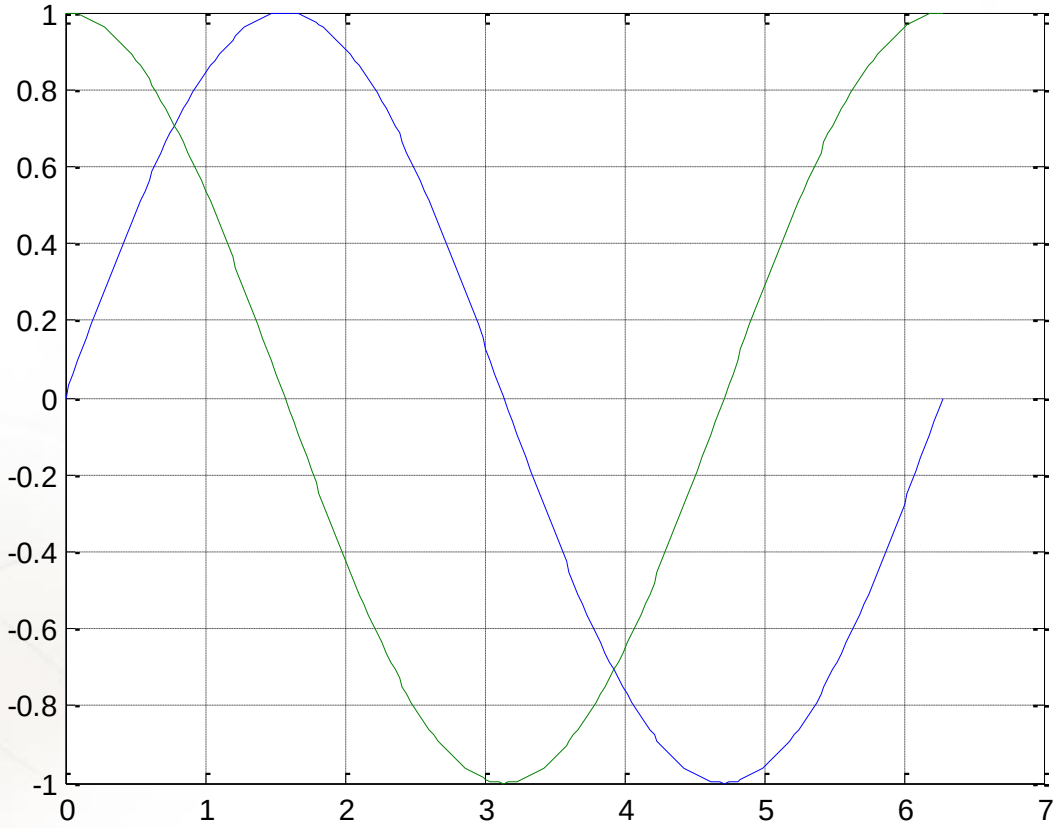
Plot

- x = 0:pi/10:6*pi;
- y = cos(x);
- plot(x,y)
- xlabel('x(radium)')
- ylabel('Sine of x')
- title('Plot of the Sine Function')



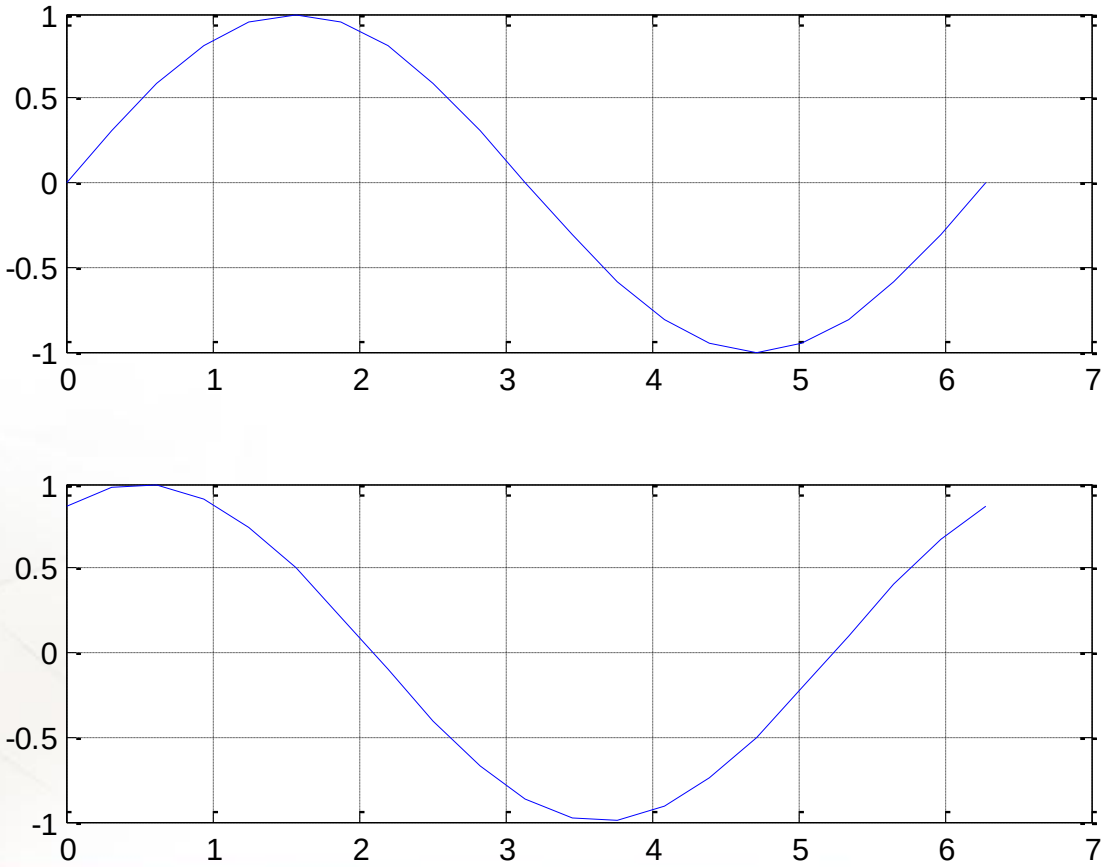
Usage of hold

```
x=0:pi/100:2*pi;
y1=sin(x);
y2=sin(x+pi/2);
plot(x,y1,'b');
hold on;
plot(x,y2,'g'),grid
```



Usage of subplot

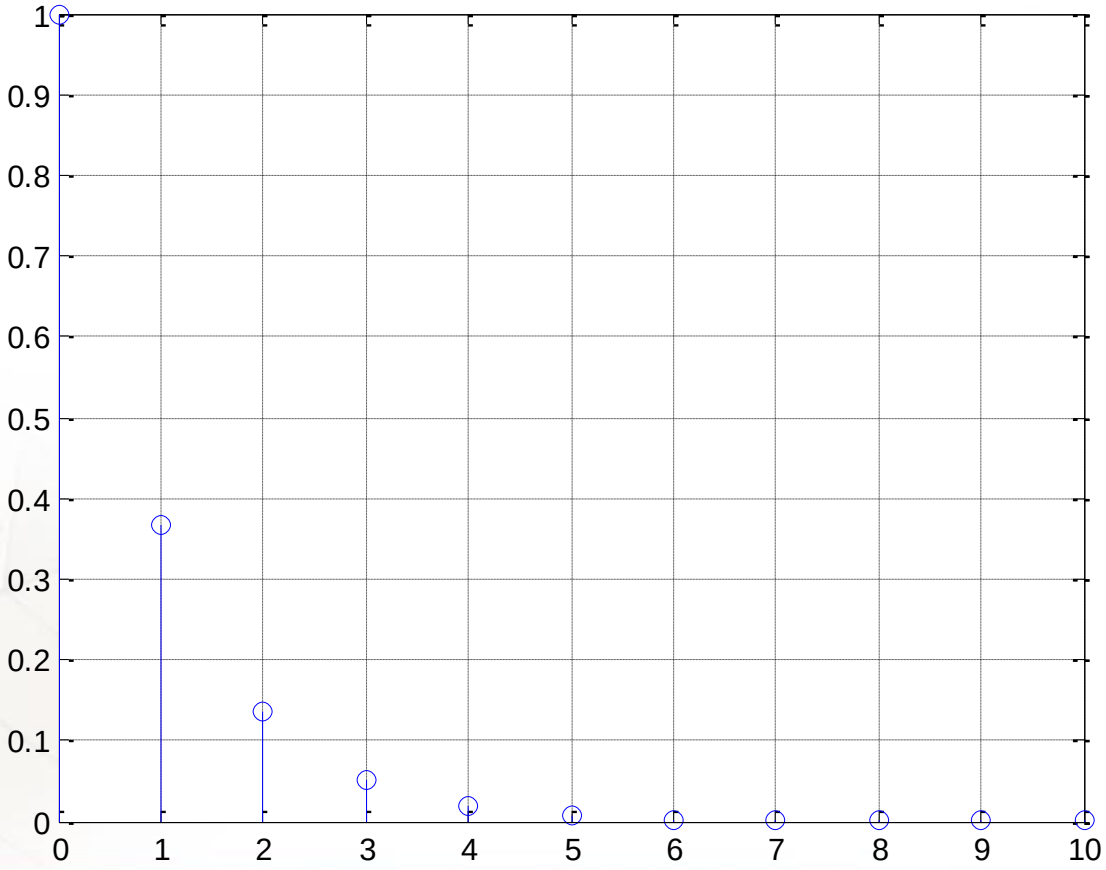
```
x=0:pi/10:2*pi;
y1=sin(x);
y2=sin(x+pi/3);
subplot(2,1,1);
plot(x,y1),grid;
subplot(2,1,2);
plot(x,y2),grid;
```



Stem

$y[n] = e^{-n}$

- n=0:10;
- y=exp(-n);
- stem(n,y),grid;



How to get help?

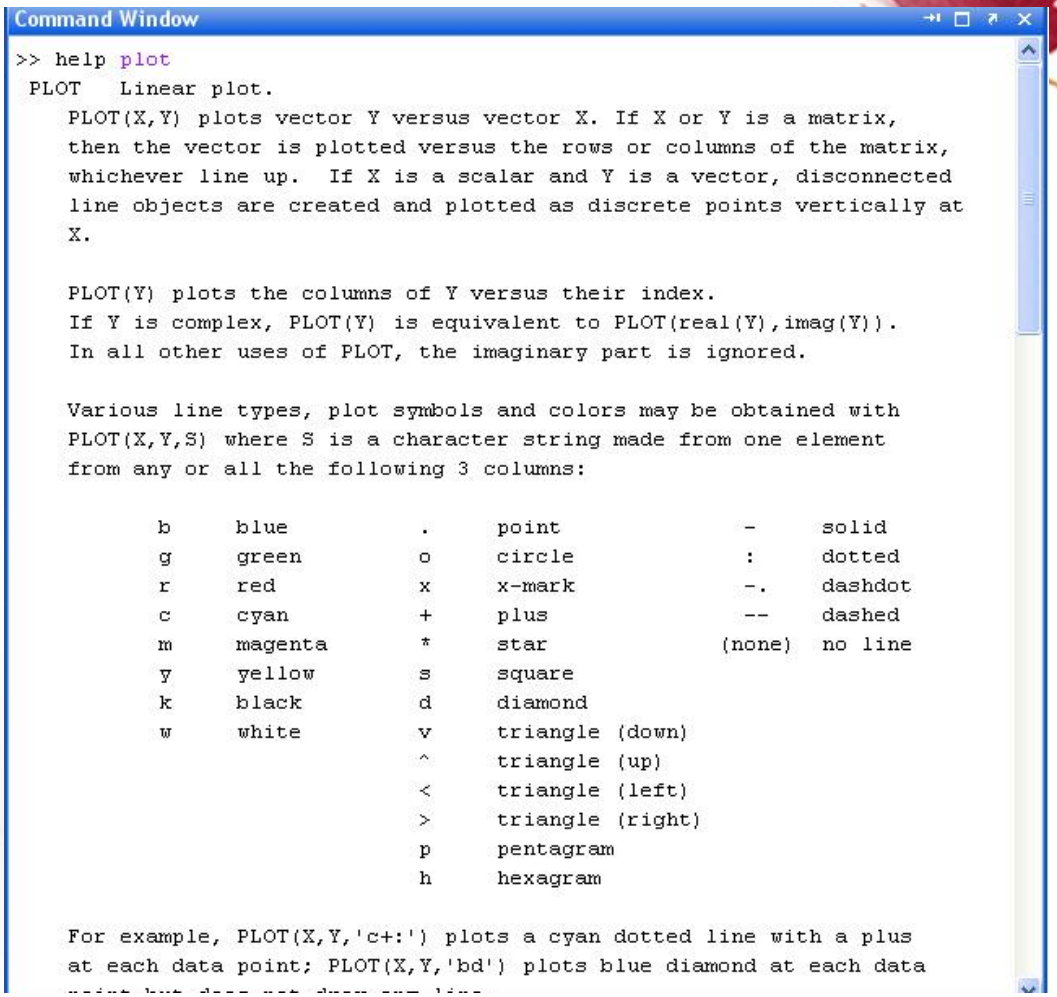
- help

Display help text

in Command Window.

e.g.,

>>help plot



Exercise

- ✓ Set up matrix A of 4x5,matrix B of 5x3;
  do multiplication C=A*B with comma and without comma behind.

- ✓ Set up two matrices of 4x3,
  do multiplication, addition, subtraction with element-by-element operaton;

- ✓ Plot x=sin(x)+cos(x+pi/3)
  specify the range of x with colon.
  the plot needs x label, y label, title, green color, grid.

Enjoy signals……