



# **Introduction to MATLAB**

# Introduction

- MatLab : **Matrix Laboratory**
- Numerical Computations with matrices
  - *Every number can be represented as matrix*
- Why Matlab?
  - User Friendly (GUI)
  - Easy to work with
  - Powerful tools for complex mathematics

# MATLAB - An Introduction

- MATLAB is both a computer programming language and software environment for using that language effectively
- MATLAB, which stands for Matrix LABoratory, is a powerful, general-purpose system or environment for doing mathematics, scientific and engineering calculations.

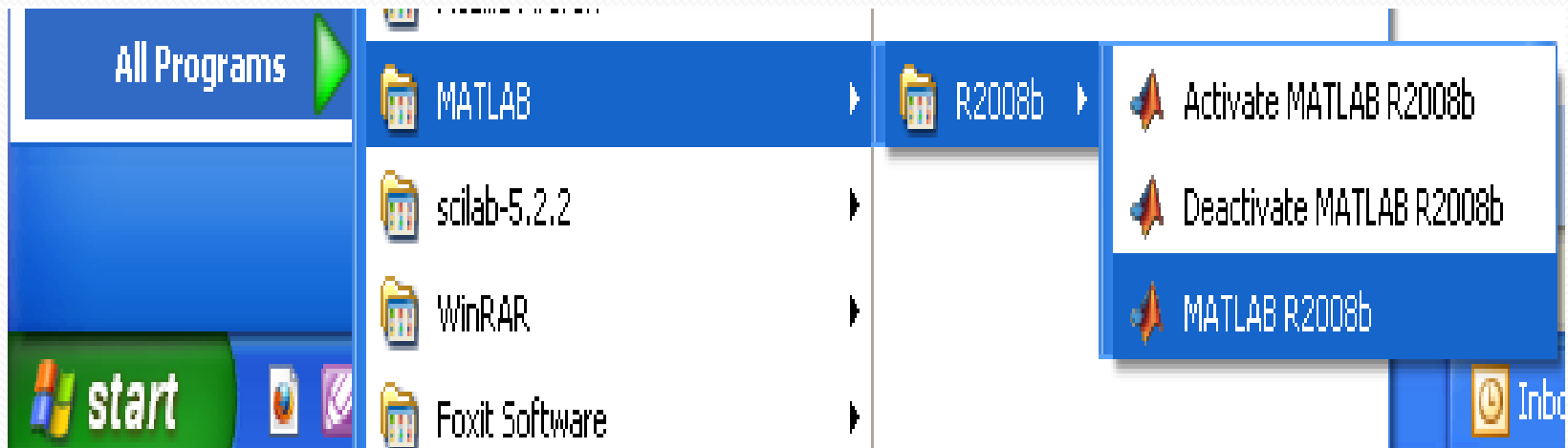
- **MATLAB is matrix-oriented**  
what can be accomplished through several statements in C or Fortran can usually be accomplished in just a few lines using MATLAB's built-in matrix and vector operations.
- **MATLAB is available for MS Windows, Macintosh personal computer, Unix and other operating systems.**
- **MATLAB is a "High-Performance Numeric Computation and Visualization Software" package.**

# Toolboxes in MATLAB

- Matlab can be augmented by a number of toolboxes and other utilities such as the Matlab compiler
- Some of the toolboxes include
  - Aerospace Toolbox
  - Image Processing Toolbox
  - Optimization Toolbox
  - Signal Processing Toolbox
  - Wavelet Toolbox
  - and so on....

# Starting the MATLAB Program

- The way you start the MATLAB program depends on the platform you use
  - On Microsoft Windows platforms
    - Start → All Programs → MATLAB → R2008b → MATLAB R2008b





Alternately, double-click the shortcut icon for MATLAB on your Windows desktop

- From DOS window, cd to the directory in which you want to start MATLAB and type matlab at the DOS prompt

# Desktop in MATLAB

- First time you start MATLAB, the desktop appears with the Default layout.
- Default Desktop comprises:
  - Current Directory
  - Command Window
  - Workspace
  - Command History



HOME PLOTS APPS

Search Documentation

FILE VARIABLE CODE SIMULINK ENVIRONMENT RESOURCES

Home: New Script, New, Open, Find Files, Compare, Import Data, Save Workspace, Clear Workspace, New Variable, Open Variable, Analyze Code, Run and Time, Clear Commands, Simulink Library, Layout, Parallel, Preferences, Set Path, Help, Add-Ons, Community, Request Support.

C:\Users\ADMIN\Documents\MATLAB

Workspace Current Folder

Name	Value	Min	Max
------	-------	-----	-----

Command History

```
edit
6x Untitled
%-- 03-03-2016 15:22 --%
[Untitled
Untitled
>9 Untitled
edit
Untitled
%-- 27-04-2016 14:20 --%
A=[1 1 0 1;1 -2 0 0;1 0 -1 2]
- det(A)
%-- 13-07-2016 22:39 --%
```

Command Window

New to MATLAB? Watch this [Video](#), see [Examples](#), or read [Getting Started](#).

fx &gt;&gt;

Ready

# Current directory

- Also known as the **Present working directory** or the **Current folder**
- You can specify the current directory for the MATLAB application when it first starts. This directory is called the **Startup directory**.
- Quick way to see the full path to the current directory or change it to a different directory is by using the **Current Directory field** in the desktop toolbar

# Command History

- Displays a log of the statements most recently run in the Command Window
  - Timestamp that marks the start of the session
  - Statements that are run in the Command Window, includes statements run using the **Evaluate Selection item** on context menus in tools such as the **Editor**, **Command History**, and **Help browser**

# Workspace

- Consists of the set of variables built up during a session of using the MATLAB software and stored in memory
- Variables are added to the workspace by using functions, running M-files, and loading saved workspaces
  - Variables used can be seen using the **who/whos** command
  - Variables can be deleted from the **Workspace** using the **Clear** command

# Command Window

- One of the main tools you use to enter data, run MATLAB functions and other M-files, and display results
- Command Window prompt, `>>`, is where you
  - Enter a MATLAB function with arguments
  - Assign values to variables, etc.
  - Browse for functions

# Editor in MATLAB

- Used to create and debug M-files, which are programs you write to run MATLAB functions
- Provides a graphical user interface for text editing, as well as for M-file debugging
- Provides color context for the code and is also a debugger.



# M – files

- Files that contain pieces of MATLAB code are called "M-files".
- Name of an M-file begins with an alphabetic character and has an extension .m.
- They are ascii text files (can be viewed using notepad)
- Two kinds of m-files:
  - **Script m-files:** Contains a batch of commands that MATLAB will execute one by one
  - **Function m-files:** Files that do specific tasks



# Executing M-files

- You can execute a M-file in the following ways:
  - Enter **run <nameofmfile>** in the **Command window**
  - To execute M-file using the Editor
    - Select **Open** from the **File** menu, browse and select the m-file
      - Alternately, click the **Open file** icon from the toolbar or press **Ctrl+O**
    - To run the entire m-file file click the **Run** icon or press **F5** key
    - Alternately, select **Evaluate Entire File** from the **Cell** menu
      - Note: This can be used only if the file is broken up into cells

# Help in MATLAB

- You can find help about MATLAB commands and functionalities by
  - Clicking the **Help** icon and browsing for the required information
  - Using **lookfor <keyword>** command
  - Using the **help** command in the Command window

# Matrix Manipulations in MATLAB

- Perform basic matrix operations
- Solve System of Linear Equations
- Perform Matrix factorization
- Compute Powers and Exponentials of matrices
- Compute Eigenvalue and Eigenvectors of matrices
- **Note:** matfun directory contains Matrix (linear algebra) functions (Enter **help matfun** in the **Command window**)

# Matrices in Matlab

- To enter a matrix

$$\begin{array}{cc} 3 & 1 \\ 6 & 4 \end{array}$$

```
>> A = [3 1 ; 6 4]
```

```
>> A = [3, 1 ; 6, 4]
```

```
>> B = [3, 5 ; 0, 2]
```

# Basic Mathematical Operations

Addition:

$$>> C = A + B$$

Subtraction:

$$>> D = A - B$$

Multiplication:

$$>> E = A * B \text{ (Matrix multiplication)}$$

$$>> E = A .* B \text{ (Element wise multiplication)}$$

## Division:

### *Left Division and Right Division*

>>  $F = A ./ B$  (Element wise division)

>>  $F = A / B$  ( $A * \text{inverse of } B$ )

>>  $F = A . \setminus B$  (Element wise division)

>>  $F = A \setminus B$  (inverse of  $A * B$ )

# MATRIX OPERATIONS

- `clc`
- `clear all`
- `A=[1 2;3 4]`
- `B=[4 5;8 12]`
- `C=A+B`
- `D=A*B`
- `E=inv(A)`
- `F=B'`
- `H=A.^2`
- `I= det(A)`

# CIRCLE

% Circle with centre(1,3)

- **`t = linspace(0, 2*pi, 101);`**
- **`x = 1 + 2*cos(t);`**
- **`y = 3 + 2*sin(t);`**
- **`plot(x,y)`**
- **`axis equal`**



# CURVE PLOT

% plotting three functions without hold on

- `x = linspace(0,1,101)`
- `plot(x,x.^3,'r+',x,sin(x),'b-',x,exp(x),'g.')`

## PLOTTING 3 FUNCTIONS-WITH HOLD ON

- `% plotting three functions with hold on`
- `clear all`
- `x = linspace(0,1,101)`
- `plot(x,x.^2,'r*')`
- `hold on`
- `plot(x,sin(x),'g.')`
- `hold on`
- `plot(x,exp(x),'b+')`

# Solve simultaneous equation

- % solve  $4x+5y=7, 7x+8y=21$
- $A=[4 \ 5; 7 \ 8]$
- $b=[7 \ 21]'$
- $x=A \setminus b$

# SOLVING QUADRATIC EQUATION

%solving quadratic equations

- `A=solve('2*x^2+5*x+12')`

# SUBPLOT

- `clc`
- `clear all`
- `x=0:.1:2*pi;`
- `subplot(2,2,1);`
- `plot(x,sin(x));`
- `subplot(2,2,2);`
- `plot(x,cos(x));`
- `subplot(2,2,3)`
- `plot(x,exp(-x));`
- `subplot(2,2,4);`
- `plot(x,sin(3*x));`