Experiment No. 1

Introduction to MATLAB and Signals in MATLAB

1.1 Objectives

1. Familiarization with MATLAB development windows and Programming fundamentals in context of MATLAB
2. Learning basic syntax and use of arithmetic operators
3. To get basic knowledge of build in commands
4. To be familiar with functions
5. To be familiar with flow control of a program

Introduction

MATLAB, short for Matrix Laboratory is a programming package specifically designed for quick and easy scientific calculations and I/O. It has literally hundreds of built-in functions for a wide variety of computations and many toolboxes designed for specific research disciplines, including statistics, optimization, solution of partial differential equations and data analysis. The purpose of this lab is to present the basics of MATLAB.

This Lab is intended for those who want to learn basics of MATLAB programming language. Even with a limited knowledge of this language a beginning programmer can write his/her own computer code for solving problems that are complex enough to be solved by other means. Numerous examples included in this text should help a reader to learn quickly basic programming tools of this language. Topics discussed include the m-files, inline functions, control flow, relational and logical operators, strings, cell arrays, rounding numbers to integers and MATLAB graphics.

Equipment

1. PC
2. Installed MATLAB

MATLAB Window Overview

Starting MATLAB

Open MATLAB by clicking on the MATLAB icon present on the desktop. The main MATLAB interface shown on the screen is called the MATLAB desktop. The desktop is a window that contains other important windows. The major tools within or accessible from the desktop are shown in the following figure.

1. Command Window: Where you enter commands.
2. Command History: Running history of commands which is preserved across sessions.
3. Current directory: Information about the current folder. Default is $matlabroot/work
4. Workspace: GUI for viewing, loading and saving MATLAB variables.

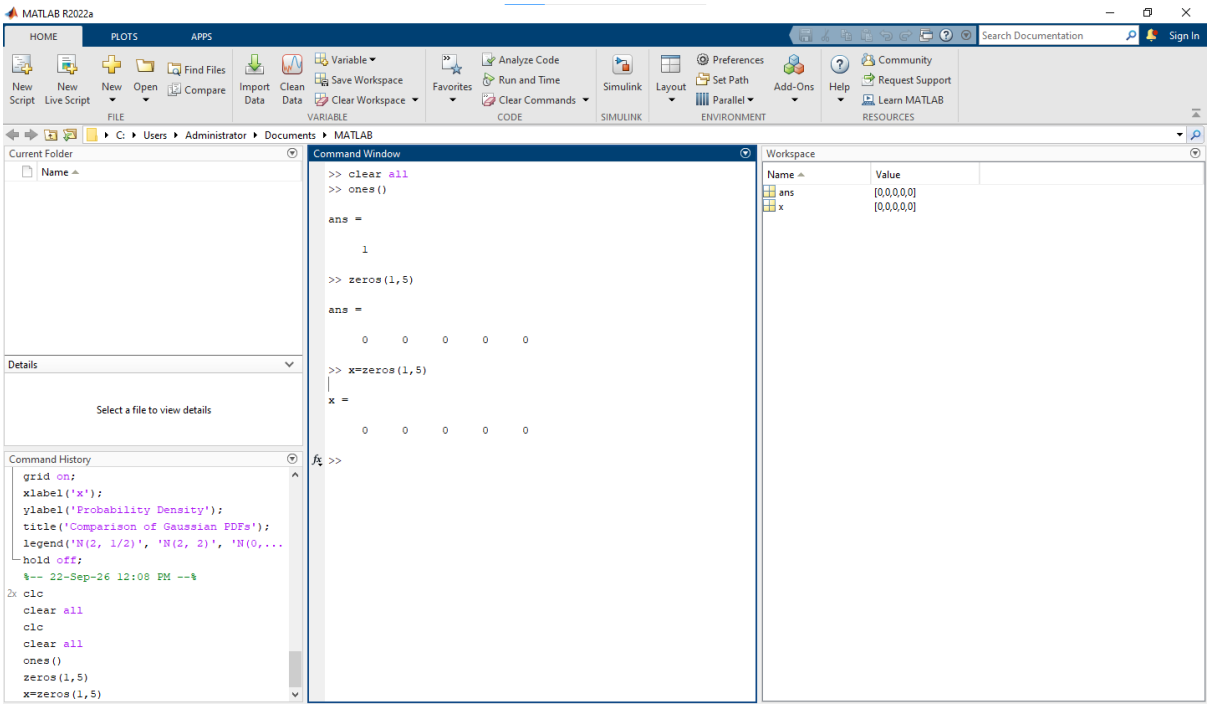


Figure 1.1: The MATLAB desktop with different windows

m-files

Files that contain a computer code are called the m-files. There are two kinds of m-files: the script files and the function files. Script files do not take the input arguments or return the output arguments. The function files may take input arguments or return output arguments.

To make the m-file click on File next select New and click on M-File from the pull-down menu. You will be presented with the MATLAB Editor/Debugger screen. Here you will type your code, can make changes, etc. Once you are done with typing, **click on File**, in the MATLAB Editor/Debugger screen and select **Save As**…. Chose a name for your file, e.g., **firstgraph.m** and click on **Save**. Make sure that your file is saved in the **directory** that is in MATLAB's search path.

To open the m-file from within the **Command Window** type **edit firstgraph.m** and then press **Enter or Return key**. There are two categories of M-files: the **Scripts and the Functions.**

Scripts

Scripts are the M-files with MATLAB commands. Their name must have a .m suffix. Scripts are suitable for solving the problems that require many commands. The advantage of the scripts is that they are implemented very easily.

Activity-1

Write a script file to plot sine wave and execute it.

```
clc          % Command to clear all the text from the Command
Window
```

```
clear       % Command to clear all variables from the current
workspace

close all   % Command to clear open MATLAB figure windows

t=-5:0.001:5;

x=sin(2*pi*t);

plot(t,x);  % Command to plot a 2-D graph
```

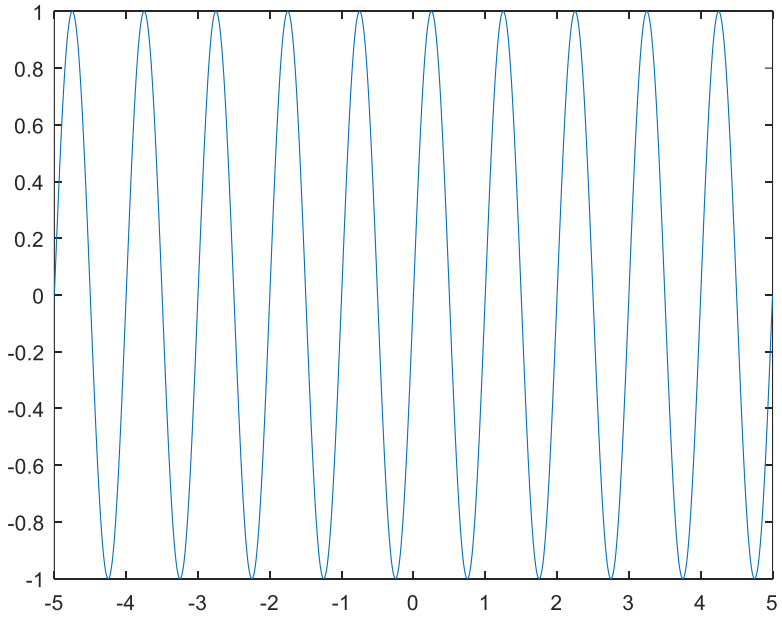


Figure 1.2: Output of Activity-1

Functions:

Function are also M-files, that is, are files with extension .m and must be saved in the current Directory of MATLAB. The difference between functions and scripts is that a function accepts one or more input arguments and returns one or more output arguments. To declare that an M-file is a function the first line of the m file must contain the syntax definition. More specifically, the first line of the M-file must be of the form:

function [output]=function_name(input)

function[y1,y2,y3,…yn]=addition{x1,x2,x3… xm}

The variable $y_1,y_2,\ldots y_n$ are the outputs of the function while $x1,x2,\ldots x_m$ are the input arguments. In case there is only one output, the square brackets are not necessary. The "name" specifies the name of the function. In order to execute a function, first the M-File is saved in **Current Directory.**

Activity-2

Write a function file to add two numbers and execute it. Function should accept two numbers as input and returns their sum and product. Also display the result on command window.

```
function [ Sum,Product ] = addition( A,B )
%Summary of this function goes here
%Detailed explanation goes here
Sum=A+B;
Product=A*B;
end
```

m-File:

```
clc
clear
close all
A=5;
B=6;
[Sum,Product]=addition(A,B);
disp('Additon Result: ');
disp(Sum)
disp('Multplication Result: ');
disp(Product)
```

Output

```
Additon Result:
    11

Multplication Result:
    30
```

Useful Commands

Here we will learn and practice useful (when working with vectors and matrices) commands.

1. The command **sum** returns the sum of the elements of a vector.
2. The command **cumsum** returns a vector whose elements are the cumulative sum of the previous elements.

3. The command **prod** is the product of the vector elements
4. The command **diff** returns a vector in which each element is given by its subtraction with the previous element.
5. The command **max** and **min** return the largest and smallest elements of the vector, respectively, as well as their index.
6. The command **sort** sorts the vector elements in ascending (by default) or descending order.
7. The command **mean** computes the mean value, while the command median returns the median value.

All these commands are suitable also for matrices by slightly changing their syntax.

Useful MATLAB Commands

Commands	Results/Comments
a = [4 2 7 0 6]	a = 4 2 7 0 6 %Definition of vector a.
s = sum(a)	s = 19 %Sum the elements of a.
c = cumsum(a)	c = 4 6 13 13 19 %Cumulative sum. the result is obtained as [4,4+2,4+2+7, 4+2+7+0, 4+2+7+0+6]
p = prod(a)	p = 0 %Product of all elements.
d = diff(a)	d = -2 5 -7 6 %Difference between two consecutive elements i.e., d(1)= a (2)- a(1), etc.
[m,i]= max(a)	m = 7, i = 3 %The largest value is assigned to variable m, and its index is assigned to variable i.
[m,i]= min(a)	m = 0, i = 4 %The smallest value is assigned to variable m, and its index is assigned to variable i.
max(a)	ans = 7 %If no output variable is specified, only the largest value is returned.
mean(a)	ans = 3.8000 %Mean value of the elements.
median(a)	ans = 4 %Median value of the vector.
sort(a)	ans = 0 2 4 6 7 %Sorting in ascending order.
sort(a,'descend')	ans = 7 6 4 2 0 %Sorting in descending order.

Special Forms of Matrices

The command **ones(M,N)** creates a matrix of size MxN with ones. Typing **ones(M)** returns a square matrix with ones. Similarly, the command **zeros(M,N)** creates a matrix of size MxN with zeros. The command **rand(M,N)** returns an MxN matrix with random elements. The

command **eye(M,N)** defines an MxN matrix with ones in the main diagonal zeros everywhere. The command **magic(M)** creates a square matrix with the property that the sum of each row or column is equal.

Useful MATLAB Commands for Matrices

Commands	**Results/Comments**
ones(2,3)	ans = 1 1 1 1 1 1 %Matrix of size 2x3 with ones.
zeros(1,4)	ans = 0 0 0 0 %Matrix of size 1x4(or vector of length) with zeros.
rand(3)	ans = 0.8600 0.9000 0.4600 0.6000 0.1000 -0.3000 0.4540 0.6023 0.2700 %Matrix of size 3x3 with random elements. If there is one input argument to the command, the obtained matrix is square.
eye(4,2)	ans = 1 0 0 1 0 0 0 0 %Magic of size 4x2 with ones in the main diagonal and zeros elsewhere.
eye(3)	ans= 1 0 0 0 1 0 0 0 1 %The identity matrix 1 of size 3x3.

Signal/Functions Plotting in MATLAB

MATLAB is a very reliable and power full tool for plotting. A graph is constructed as a set of points in two or three dimensions. These points are typically connected with a solid line. Commonly a graph of a function is required. However, in MATLAB a plot is done using the vectors and matrices not functions.

Plotting in Two Dimensions

Suppose that we want to plot a signal (t)=cos(2πft), where t is the value of time. The procedure to plot (t) is as follows:

1. Vector t is created. Such as $a \leq t \leq b$, where a and b are scalars.
2. The function (t) will be plotted over the interval $[a, b]$.
3. Create the signal (t).
4. The value of each element of y is the value of (t) calculated for each value of t.
5. Finally, the function $y(x)$ is plotted by typing 'plot(x,y)'

Activity-3

Write a script file to plot cosine wave of 1Hz frequency.

```
clc
clear
close all
t=0:0.001:1;
f=1;
y=cos(2*pi*f*t);


% Plot the signal versus time:
plot(t,y);
grid on;
```

Plotting multiple functions/signals in one figure with proper formatting

It is possible to plot more than one function in the name of figure by employing a different syntax of command Plot. The command grid on adds lines to the graph, while the command grid off removes the grid lines. Simply typing grid is switch between the two modes. Text besides the x-axis and y-axis can be added using the commands xlabel and ylabel, respectively. A graph title is inserted by the Command title. In the previous examples, the functions were plotted with predefined colors and line type (solid). It is possible to create a graph using colors, symbols used to draw the points, and type of line that connects the points of your choice. This is achieved by applying one more input argument to the command Plot. The new argument is a series of special character given in single quotes. The available special characters are presented in the table below.

Activity-4

Write a script file to plot cosine and sine wave of 1Hz frequency in same figure with proper formatting.

```
clc
clear
close all
t=0:0.001:1;
```

```
f=1;
y1=cos(2*pi*f*t);
y2=sin(2*pi*f*t);
% Plot the signals
plot(t,y1,'red',t,y2,'green');
xlabel('time (seconds)');
ylabel('Amplitude');
title('Sine and Cosine Signal');
legend('Cosine','Sine');
grid on;
```

Discrete time signals/functions plotting

A discrete time function is a function of the form $f[n]$, $n \in z$, where z *denotes* the *set* of integers. In this case the appropriate command for plotting the function is 'stem(n,y)'

Activity-5

Plot the discrete function $f(n) = n^2$, where $-3 \leq n \leq 3$. Also compare it with the result if we use "plot" command.

```
clc
clear
close all
n = -3:3;
y= n.^2;
plot (n,y);
hold on
stem(n,y);
xlabel('Time Axis');
ylabel('Amplitude');
title('Graph of f(n) vs Graph of f(t)');
legend('f(t)','f(n)');
```

Lab Tasks (Psychomotor)

1. Form the matrix given below.

$$A = \begin{bmatrix} 1 & 2 & 5 \\ 4 & 8 & 2 \end{bmatrix}$$

2. Write the following matrix in MATLAB and perform the given operations.

$$A = \begin{bmatrix} 12 & 15 & 16 \\ 23 & 26 & 27 \\ 31 & 40 & 41 \end{bmatrix} \qquad B = \begin{bmatrix} 22 & 35 & 56 \\ 63 & 66 & 78 \\ 24 & 30 & 46 \end{bmatrix}$$

Find out:

1. A+B
2. B*A
3. A.*B
4. B - A

3. Plot the following function. Properly label and format the graph.

$$f(t) = \begin{cases} 1 & -2 \le t \le 2 \\ 0 & 2 < t < 5 \\ tsin(4\pi t) & 5 \le t \le 8 \end{cases}$$

4. Write a function **file** to add, subtract and multiply two numbers and execute it. Numbers should be taken as input from the user in an m-file. Call the function from m-file and display the results on command window.

Lab Tasks (Cognitive)

Q. No.	Cognitive Assessment Question	Marks
1	Given the matrix A = [[1,2,5],[4,8,2]], determine its dimensions and the total number of elements.	2
2	Given A = [[12,15,16],[23,26,27],[31,40,41]] and B = [[22,35,56],[63,66,78],[24,30,46]], determine the (2,3) element of A+B and B−A.	2
3	For f(t)=1 for −2 ≤ t ≤ 2, 0 for 2 < t < 5, and tsin(4πt) for 5 ≤ t ≤ 8, determine f(0), f(3), and f(6).	2
4	A signal x(t)=sin(2πft) has frequency 5 Hz. Determine its period and the number of cycles completed in 2 seconds.	2
5	Two numbers are a=6 and b=4. Applying the required arithmetic operations, determine a+b, a−b, and a*b.	2

Lab Exercise and Summary

Summary should cover Introduction, Procedure, Data Analysis and Evaluation.

Student's Signature: _____ **Date: _____**

LABORATORY SKILLS ASSESSMENT (Cognitive)

Obtained Marks	/10

Instructor's Signature: _____ Date: _____

LABORATORY SKILLS ASSESSMENT (Psychomotor)

Psychomotor Activity 1 — Experimental Setup

S. No	Criteria	Performance	Description	Marks
1	**MATLAB Environment Setup (4)**	Level 4	Correctly opens MATLAB, prepares the Command Window/Workspace and working directory, and is ready for the assigned programming activity.	4
		Level 3	Correctly prepares MATLAB with minor errors or assistance.	3
		Level 2	Partially prepares the required environment and requires considerable assistance.	2
		Level 1	Shows limited/incomplete ability to prepare MATLAB and requires substantial assistance.	1
2	**Script/Function File Setup (4)**	Level 4	Creates and organizes the required .m script/function with appropriate input/output structure for the assigned task.	4
		Level 3	Completes the file setup with minor errors or assistance.	3
		Level 2	Partially completes the file setup and requires considerable assistance.	2
		Level 1	File setup is largely incomplete and requires substantial assistance.	1
3	**Input & Parameter Setup (2)**	Level 2	Correctly defines signal, numerical inputs, time vector and required parameters.	2
		Level 1	Correctly defines signal, numerical inputs, time vector and required parameters. Partially completed; considerable assistance.	1
Total Obtained				/10

If a student is absent:

Criteria	Description	Marks
Absent	Student is absent from the laboratory activity.	**0**

Psychomotor Activity 2 — Implementation + Results

S. No	Criteria	Performance	Description	Marks
1	MATLAB Programming Implementation & Safe Working Practice (4)	Level 4	Correctly implements the assigned script/function, arithmetic operations and statements independently and follows proper laboratory practices.	4
		Level 3	Implements the assigned work with minor errors or assistance and generally follows proper laboratory practices.	3
		Level 2	Partially implements the assigned work and requires considerable assistance or repeated guidance.	2
		Level 1	Implementation is largely incomplete, requires substantial assistance, or shows poor adherence to proper laboratory practices.	1
2	Execution & Verification (2)	Level 2	Executes the MATLAB program successfully and checks the generated outputs.	2
		Level 1	Performs basic execution and verification with minor errors or partially completed and considerable assistance	1
3	Signal/Program Results (4)	Level 4	Produces correct plots, numerical results, waveforms with appropriate labels, or simulation outputs and appropriately verifies the required signals/system behavior.	4
		Level 3	Produces mostly correct results/plots with minor errors or omissions.	3
		Level 2	Produces partially correct results/plots with noticeable errors or omissions.	2
		Level 1	Results/plots are largely incorrect or incomplete.	1
	Total Obtained			/10

If a student is absent:

Criteria	Description	Marks
Absent	Student is absent from the laboratory activity.	**0**

Instructor's Signature: _____ **Date:** _____

LABORATORY SKILLS ASSESSMENT (Affective)

Affective Activity — Lab Report + Viva Voce

S. No	Criteria	Performance	Description	Marks
1	**Lab Report Completeness & Organization (4)**	Level 4	Report is complete, well-organized, properly presented, and demonstrates responsible completion of the assigned Signals and Systems laboratory work.	**4**
		Level 3	Report is complete with minor omissions or organizational issues and demonstrates satisfactory responsibility for the assigned work.	**3**
		Level 2	Report is partially complete with several omissions and demonstrates limited responsibility for the assigned work.	**2**
		Level 1	Report has major omissions or poor organization and demonstrates minimal responsibility for the assigned work.	**1**
2	**Code, Results & Conclusion (2)**	Level 2	MATLAB code, results/plots, and conclusion are correctly presented and discussed, demonstrating ownership of the assigned work.	**2**
		Level 1	MATLAB code, results/plots, or conclusion are presented with minor errors or omissions.	**1**
3	**Viva Voce (4)**	Level 4	Clearly explains the assigned MATLAB code, signal/system concepts, analysis, results, and plots, demonstrating individual responsibility, participation, and effective communication.	**4**
		Level 3	Explains most of the assigned work and results correctly, with minor difficulty, and demonstrates good individual participation.	**3**
		Level 2	Explains basic aspects of the assigned work with some prompting and demonstrates basic participation.	**2**
		Level 1	Demonstrates limited understanding or involvement and requires substantial prompting to explain the assigned work.	**1**
	Total Obtained			/10

The student has not submitted the report and has not appeared in the Viva Voce.

Criteria	Description	Marks
Lab Report and Results Not Submitted	Student has not submitted the report and results file.	**0**
Absent in Viva Voce	Student is absent from the viva voce.	**0**

Instructor's Signature: _____ **Date:** _____