

كلية المستقبل الاهلية الجامعة
قسم الفيزياء الطبية

MATLAB

Lec-4

اعداد
م.م علا علي عبيد

MATLAB has an excellent set of graphic tools. Plotting a given data set or the results

of computation is possible with very few commands. You are highly encouraged to plot

mathematical functions and results of analysis as often as possible. Trying to understand

mathematical equations with graphics is an enjoyable and very efficient way of learning mathematics. Being able to plot mathematical functions and data freely is the most important step, and this section is written to assist you to do just that.

2.2.2 Creating simple plots

The basic MATLAB graphing procedure, for example in 2D, is to take a vector of xcoordinates, $x = (x_1, \dots, x_N)$, and a vector of y-coordinates, $y = (y_1, \dots, y_N)$, locate the

points $(x_i$

, $y_i)$, with $i = 1, 2, \dots, n$ and then join them by straight lines. You need to prepare

x and y in an identical array form; namely, x and y are both row arrays or column arrays of

the same length.

The MATLAB command to plot a graph is `plot(x,y)`. The vectors $x = (1, 2, 3, 4, 5, 6)$

and $y = (3, -1, 2, 4, 5, 1)$ produce the picture shown in Figure 2.1.

```
>> x = [1 2 3 4 5 6];
```

```
>> y = [3 -1 2 4 5 1];
```

```
>> plot(x,y)
```

Note: The plot functions has different forms depending on the input arguments. If y is a

vector `plot(y)` produces a piecewise linear graph of the elements of y versus the index of the

elements of y . If we specify two vectors, as mentioned above, `plot(x,y)` produces a graph

of y versus x .

For example, to plot the function $\sin(x)$ on the interval $[0, 2\pi]$, we first create a vector of

x values ranging from 0 to 2π , then compute the sine of these values, and finally plot the

result:

Notes:

- `0:pi/100:2*pi` yields a vector that
 - starts at 0
 - , – takes steps (or increments) of $\pi/100$,
 - stops when 2π is reached.
- If you omit the increment, MATLAB automatically increments by

2.2.3 Adding titles, axis labels, and annotations

MATLAB enables you to add axis labels and titles. For example, using the graph from the previous example, add an x - and y -axis

labels. Now label the axes and add a title. The character `\pi` creates the symbol π . An example of 2D plot is shown in Figure 2.2.

```
>> xlabel('x = 0:2\pi')
>> ylabel('Sine of x')
>> title('Plot of the Sine function')
```

The color of a single curve is, by default, blue, but other colors are possible. The desired color is indicated by a third argument. For example, red is selected by `plot(x,y,'r')`. Note the single quotes, `' '`, around `r`

Multiple data sets in one plot :

Multiple (x, y) pairs arguments create multiple graphs with a single call to `plot`. For example, these statements plot three related functions of x : $y_1 = 2 \cos(x)$, $y_2 = \cos(x)$, and $y_3 = 0.5 * \cos(x)$, in the interval $0 \leq x \leq 2\pi$.

```
>> x = 0:pi/100:2*pi;
>> y1 = 2*cos(x);
>> y2 = cos(x);
>> y3 = 0.5*cos(x);
>> plot(x,y1,'--',x,y2,'-',x,y3,':')
>> xlabel('0 \leq x \leq 2\pi')
```

```
>> ylabel('Cosine functions')
>> legend('2*cos(x)', 'cos(x)', '0.5*cos(x)')
>> title('Typical example of multiple plots')
>> axis([0 2*pi -3 3])
```

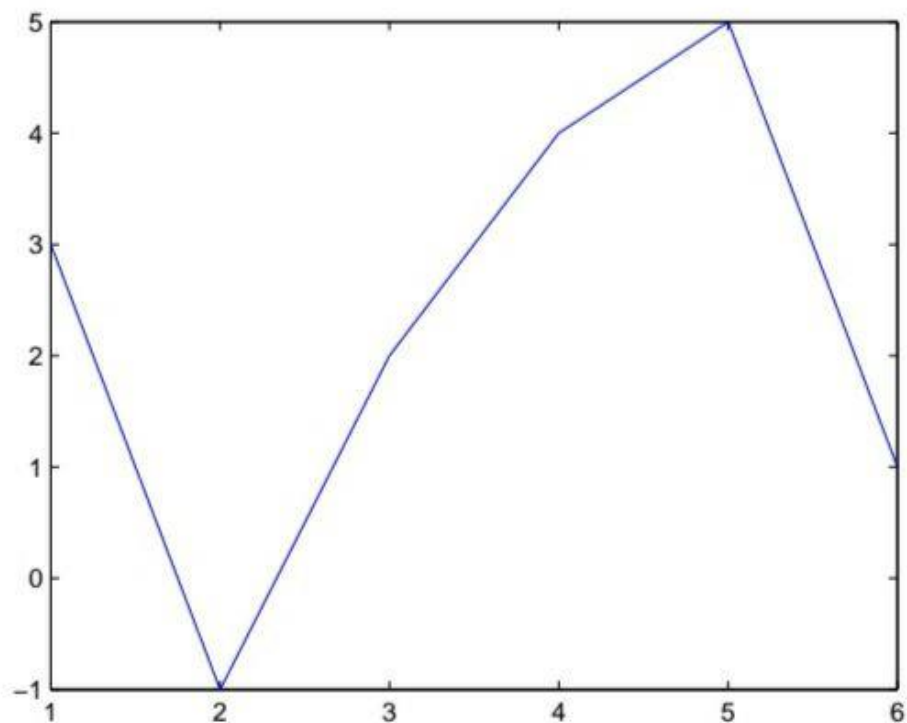


Figure 2.1: Plot for the vectors x and y

Specifying line styles and colors It is possible to specify line styles, colors, and markers (e.g., circles, plus signs, . . .) using the plot command:

```
plot(x,y,'style_color_marker')
```

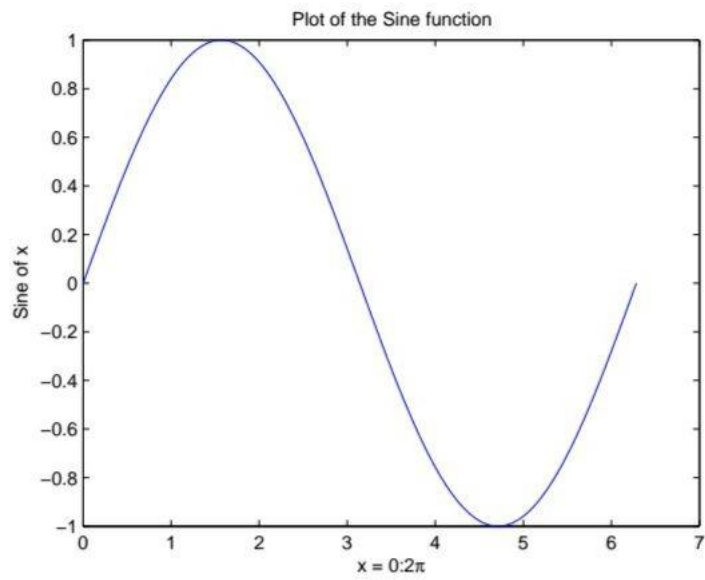


Figure 2.2: Plot of the Sine function

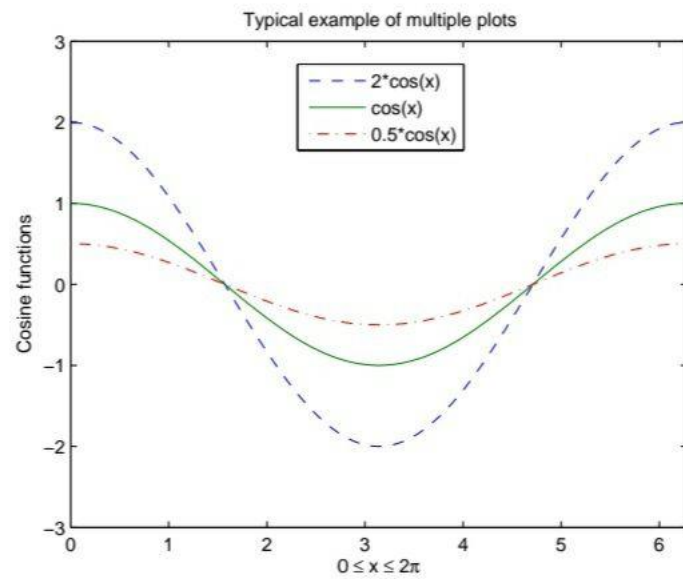


Figure 2.3: Typical example of multiple plots

