

MATLAB - Lecture # 5

Two Dimensional Plots / Chapter 5

Topics Covered:

1. Plotting basic 2-D plots.

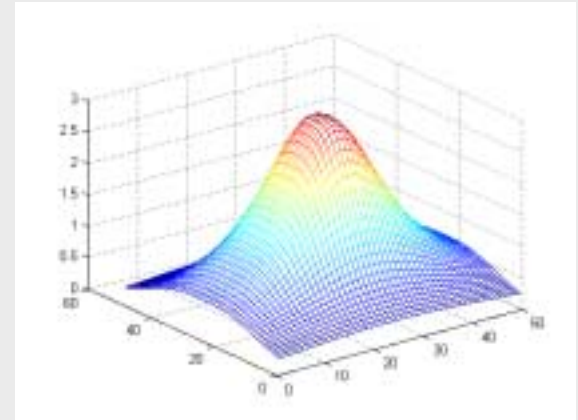
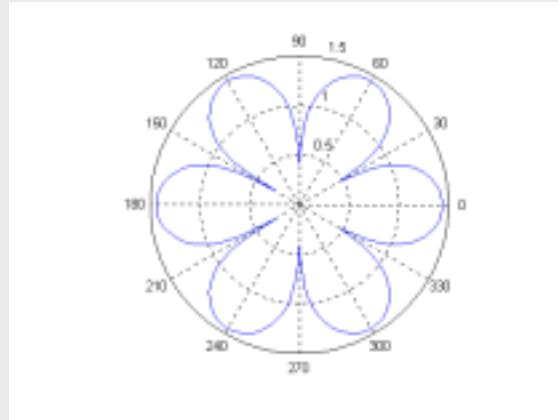
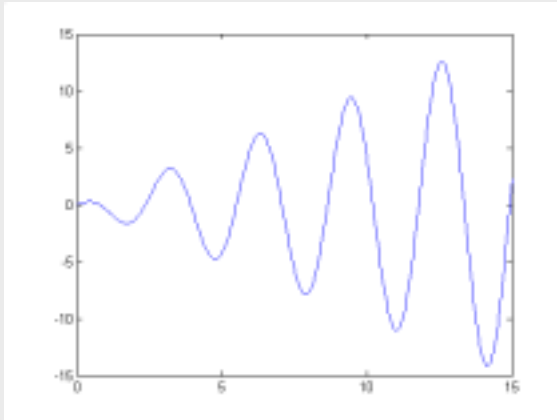
The **plot** command.

The **fplot** command.

Plotting multiple graphs in the same plot.

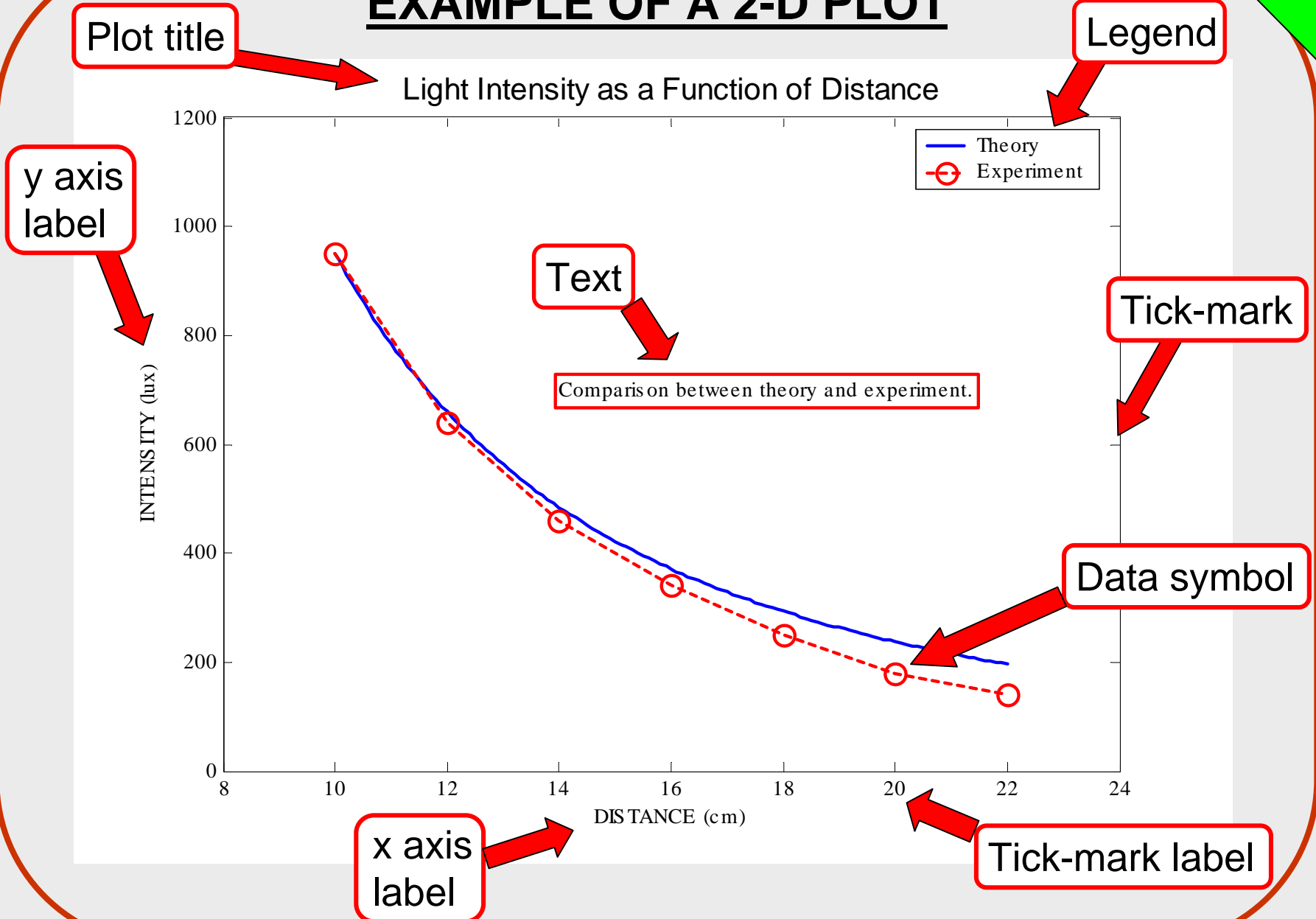
MAKING X-Y PLOTS

MATLAB has many functions and commands that can be used to create various types of plots.



In our class we will only create two dimensional x – y plots.

EXAMPLE OF A 2-D PLOT



TWO-DIMENSIONAL PLOT COMMAND

The basic 2-D plot command is:

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

where **x** is a vector (one dimensional array), and **y** is a vector. Both vectors **must** have the same number of elements.

- ❖ The plot command creates a single curve is with the **x** values on the abscissa (horizontal axis) and the **y** values on the ordinate (vertical axis).
- ❖ The curve is made from segments of lines that connect the points that are defined by the **x** and **y** coordinates of the elements in the two vectors.

CREATING THE X AND Y VECTORS

- ❖ If data is given, the information is entered as the elements of the vectors \mathbf{x} and \mathbf{y} .
- ❖ If the values of \mathbf{y} are determined by a function from the values of \mathbf{x} , then a vector \mathbf{x} is created first, and then the values of \mathbf{y} are calculated for each value of \mathbf{x} . The spacing (difference) between the elements of \mathbf{x} must be such that the plotted curve will show the details of the function.

PLOT OF GIVEN DATA

Given data:

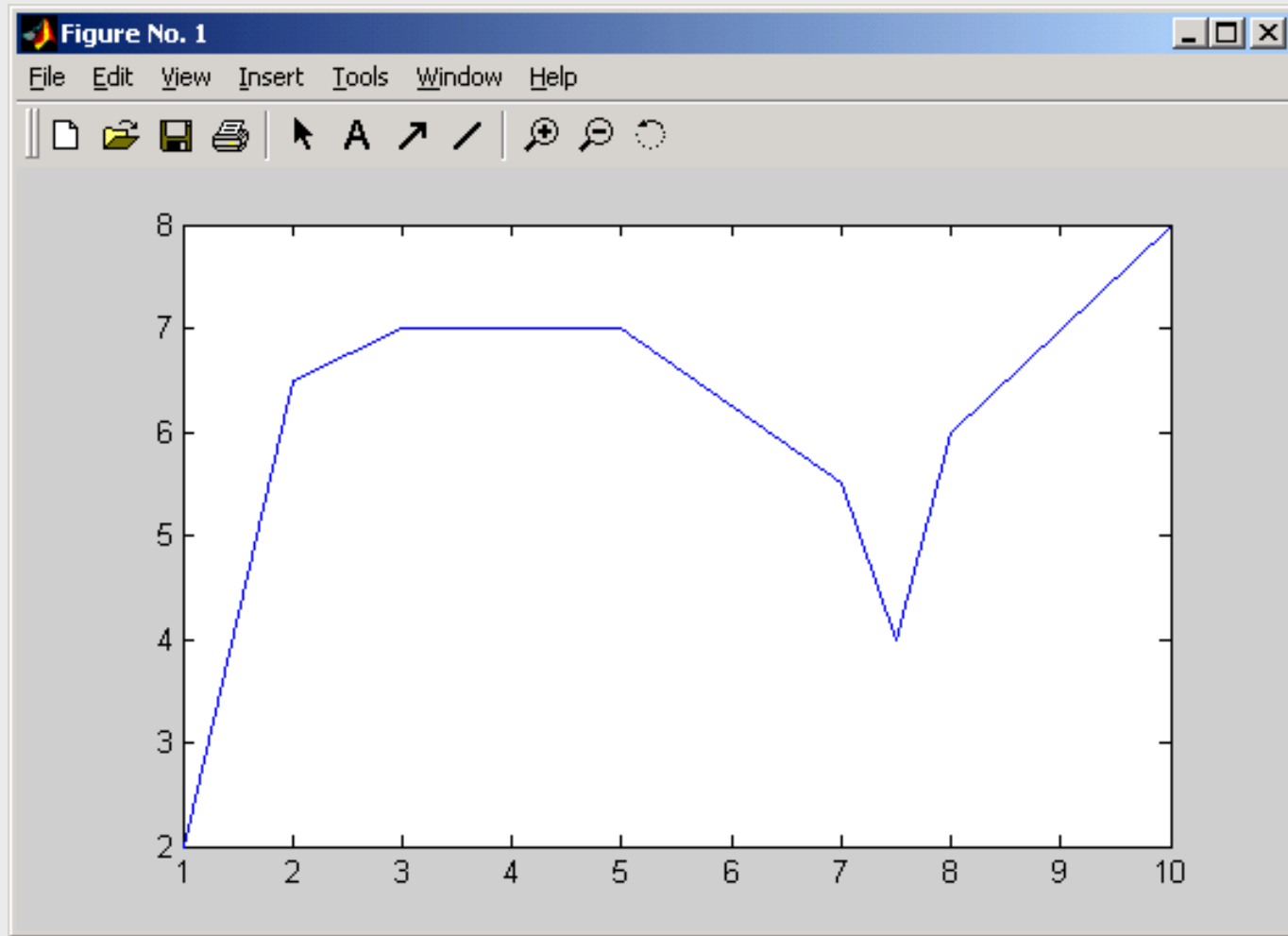
x	1	2	3	5	7	7.5	8	10
y	2	6.5	7	7	5.5	4	6	8

A plot can be created by the commands shown below. This can be done in the Command Window, or by writing and then running a script file.

```
>> x=[1 2 3 5 7 7.5 8 10];  
>> y=[2 6.5 7 7 5.5 4 6 8];  
>> plot(x,y)
```

Once the plot command is executed, the Figure Window opens with the following plot.

PLOT OF GIVEN DATA



LINE SPECIFIERS IN THE PLOT COMMAND

Line specifiers can be added in the **plot** command to:

- Specify the style of the line.
- Specify the color of the line.
- Specify the type of the markers (if markers are desired).

```
plot(x,y,'line specifiers')
```

LINE SPECIFIERS IN THE PLOT COMMAND

`plot(x,y,'line specifiers')`

Line Style	Specifier	Line Color	Specifier	Marker Type	Specifier
Solid	-	red	r	plus sign	+
dotted	:	green	g	circle	o
dashed	--	blue	b	asterisk	*
dash-dot	-.	Cyan	c	point	.
		magenta	m	square	s
		yellow	y	diamond	d
		black	k		

LINE SPECIFIERS IN THE PLOT COMMAND

- The specifiers are typed inside the plot command as strings.
- Within the string the specifiers can be typed in any order.
- The specifiers are optional. This means that none, one, two, or all the three can be included in a command.

EXAMPLES:

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

A solid blue line connects the points with no markers.

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

A red solid line connects the points.

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

A yellow dashed line connects the points.

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

The points are marked with * (no line between the points.)

```
plot(x,y,'g:d')
```

A green dotted line connects the points which are marked with diamond markers.

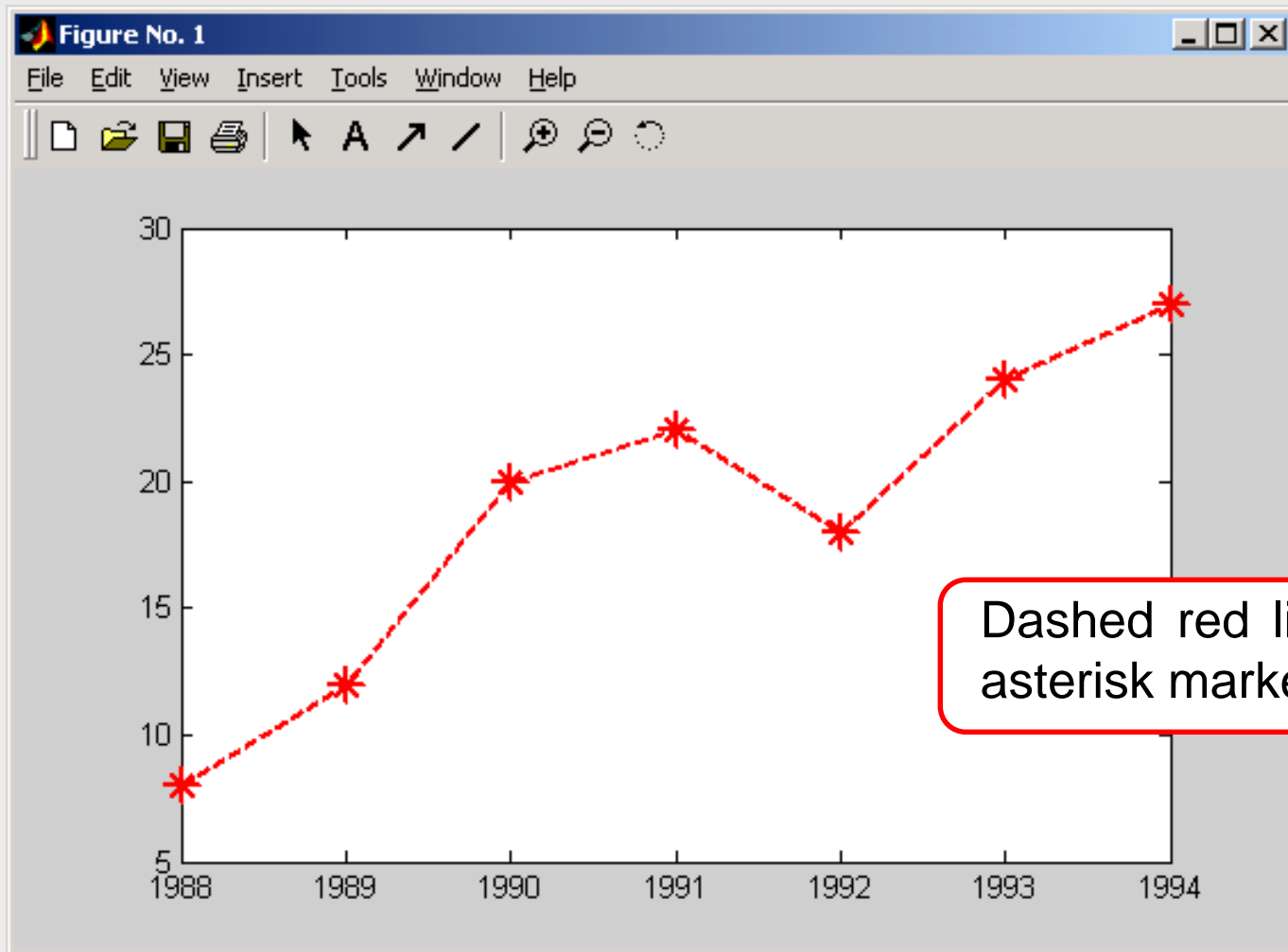
PLOT OF GIVEN DATA USING LINE SPECIFIERS IN THE `plot` COMMAND

Year	1988	1989	1990	1991	1992	1993	1994
Sales (M)	127	130	136	145	158	178	211

```
>> yr = [1988:1:1994];  
>> sle = [127, 130, 136, 145, 158, 178, 211];  
>> plot(yr,sle,'--r*')
```

Line Specifiers:
dashed red line and
asterisk markers.

PLOT OF GIVEN DATA USING LINE SPECIFIERS IN THE `plot` COMMAND



CREATING A PLOT OF A FUNCTION

Consider: $y = 3.5^{-0.5x} \cos(6x)$ for $-2 \leq x \leq 4$

A script file for plotting the function is:

```
% A script file that creates a plot of
```

```
% the function: 3.5^(-0.5x)*cos(6x)
```

```
x = [-2:0.01:4];
```

← Creating a vector with spacing of 0.01.

```
y = 3.5.^(-0.5*x).*cos(6*x);
```

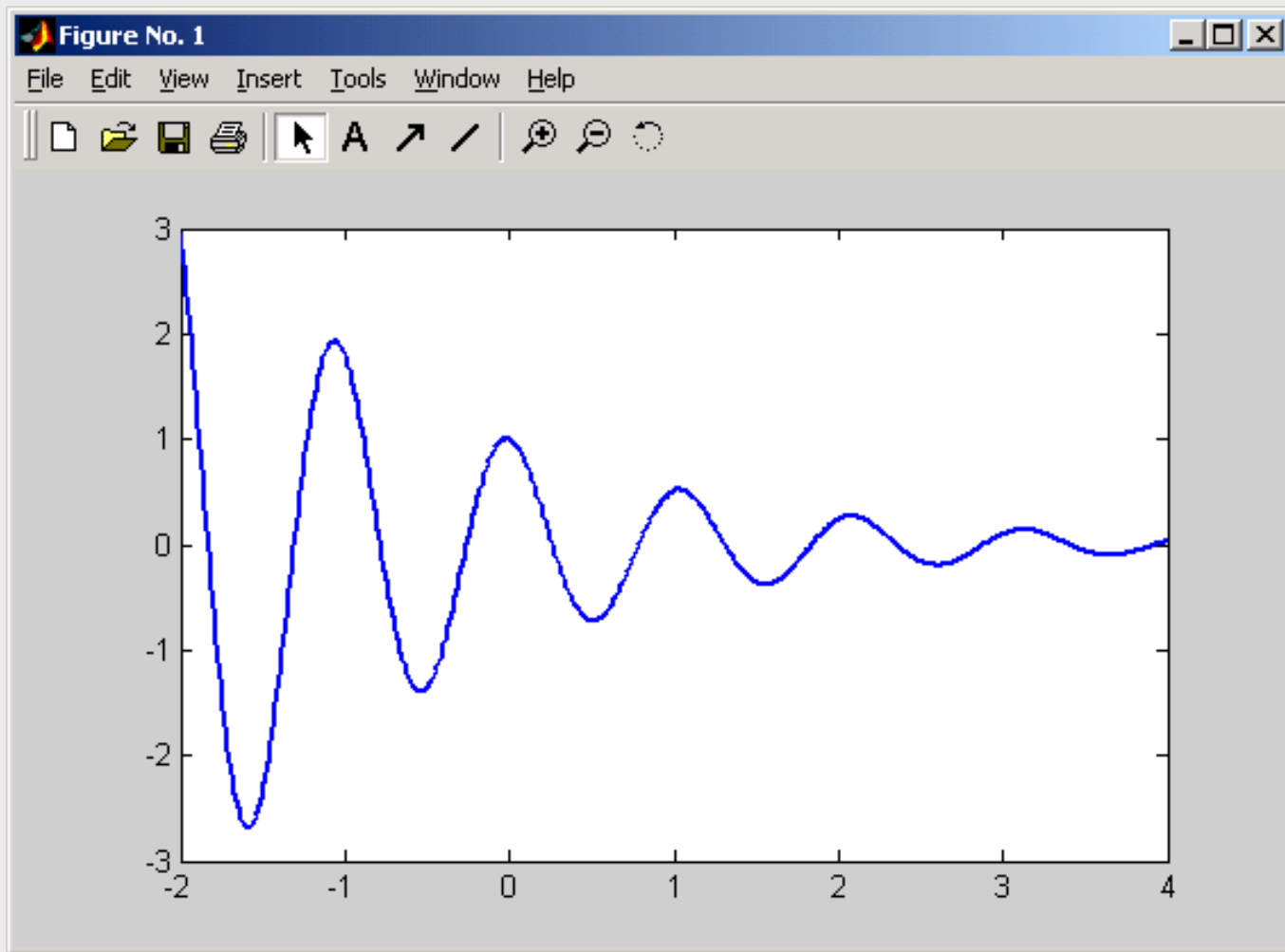
← Calculating a value of y for each x .

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

Once the plot command is executed, the Figure Window opens with the following plot.

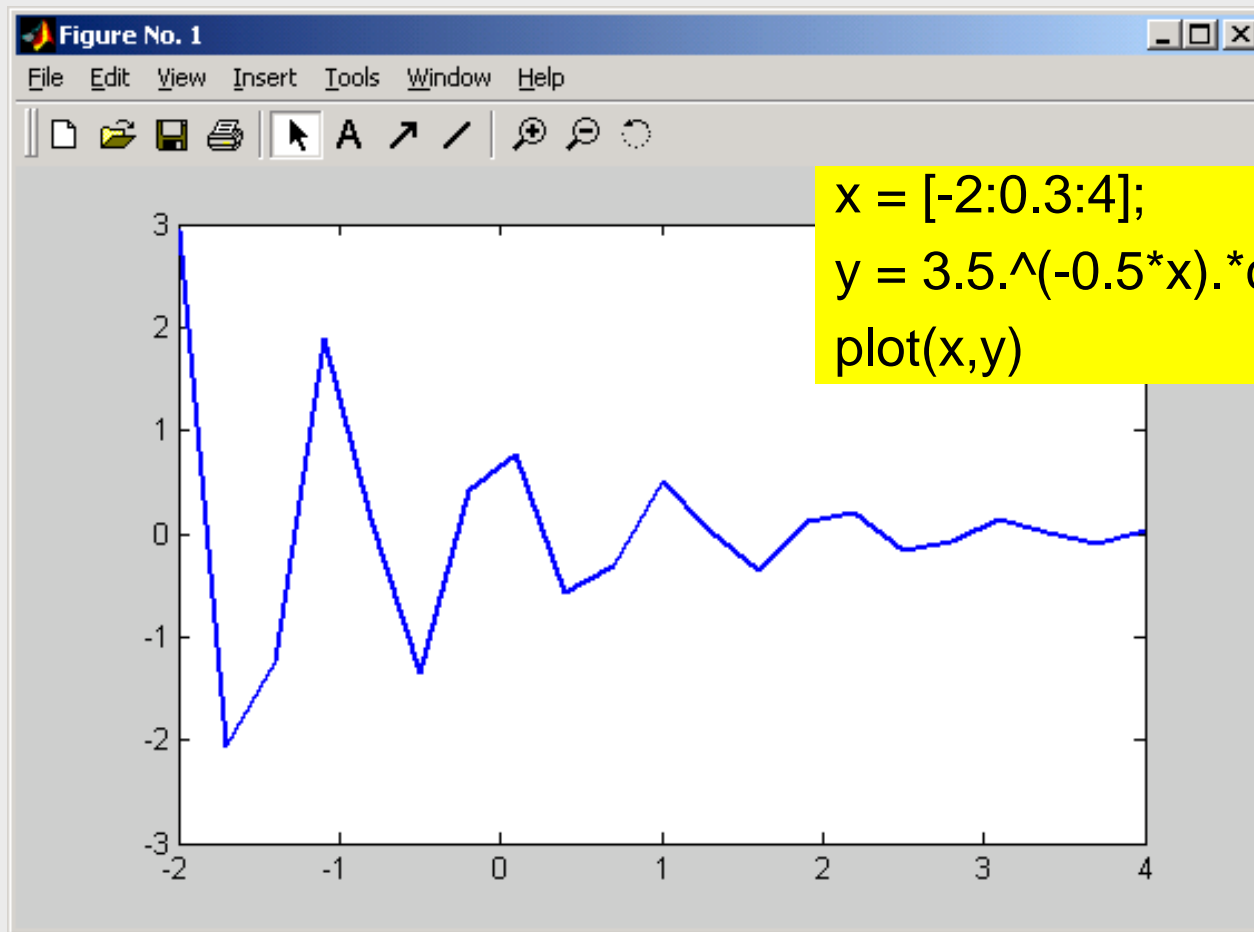
A PLOT OF A FUNCTION

$$y = 3.5^{-0.5x} \cos(6x) \quad \text{for } -2 \leq x \leq 4$$



CREATING A PLOT OF A FUNCTION

If the vector \mathbf{x} is created with large spacing, the graph is not accurate. Below is the previous plot with spacing of 0.3.



THE `fplot` COMMAND

The `fplot` command can be used to plot a function with the form: $y = f(x)$

```
fplot('function', limits)
```

- The function is typed in as a string.
- The limits is a vector with the domain of x , and optionally with limits of the y axis:

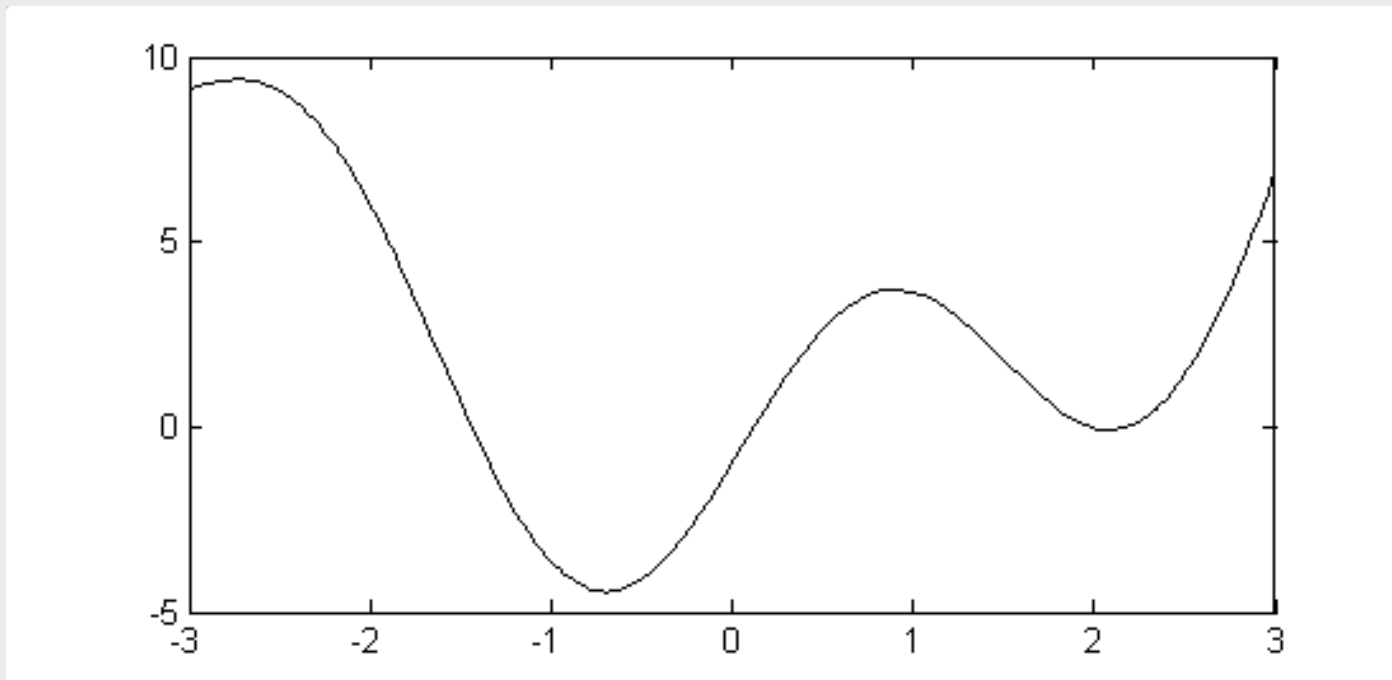
```
[xmin, xmax] or [xmin, xmax, ymin, ymax]
```

- Line specifiers can be added.

PLOT OF A FUNCTION WITH THE `fplot` COMMAND

A plot of: $y = x^2 + 4\sin(2x) - 1$ for $-3 \leq x \leq 3$

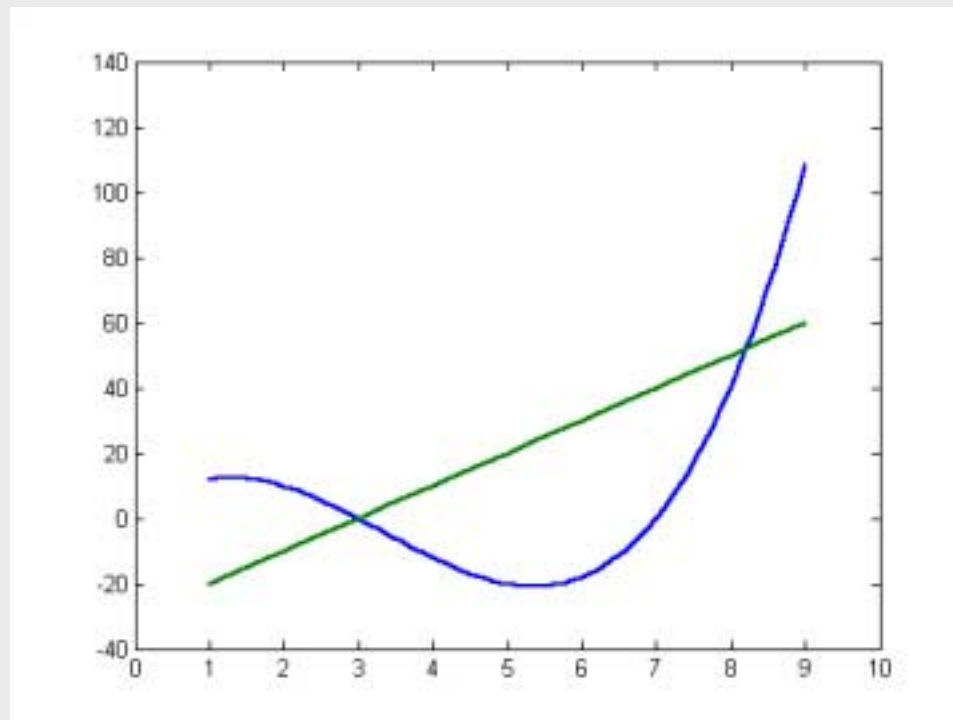
```
>> fplot('x^2+4*sin(2*x)-1',[-3 3])
```



PLOTTING MULTIPLE GRAPHS IN THE SAME PLOT

Plotting two (or more) graphs in one plot:

1. Using the **plot** command.
2. Using the **hold on**, **hold off** commands.



USING THE PLOT COMMAND TO PLOT MULTIPLE GRAPHS IN THE SAME PLOT

```
plot(x,y,u,v,t,h)
```

Plots three graphs in the same plot:

y versus **x**, **v** versus **u**, and **h** versus **t**.

- By default, MATLAB makes the curves in different colors.
- Additional curves can be added.
- The curves can have a specific style by adding specifiers after each pair, for example:

```
plot(x,y,'-b',u,v,'-r',t,h,'g:')
```

USING THE PLOT COMMAND TO PLOT MULTIPLE GRAPHS IN THE SAME PLOT

Plot of the function, $y = 3x^3 - 26x + 10$ and its first and second derivatives, for $-2 \leq x \leq 4$, all in the same plot.

```
x = [-2:0.01:4];
```

← vector **x** with the domain of the function.

```
y = 3*x.^3-26*x+6;
```

← Vector **y** with the function value at each **x**.

```
yd = 9*x.^2-26;
```

← Vector **yd** with values of the first derivative.

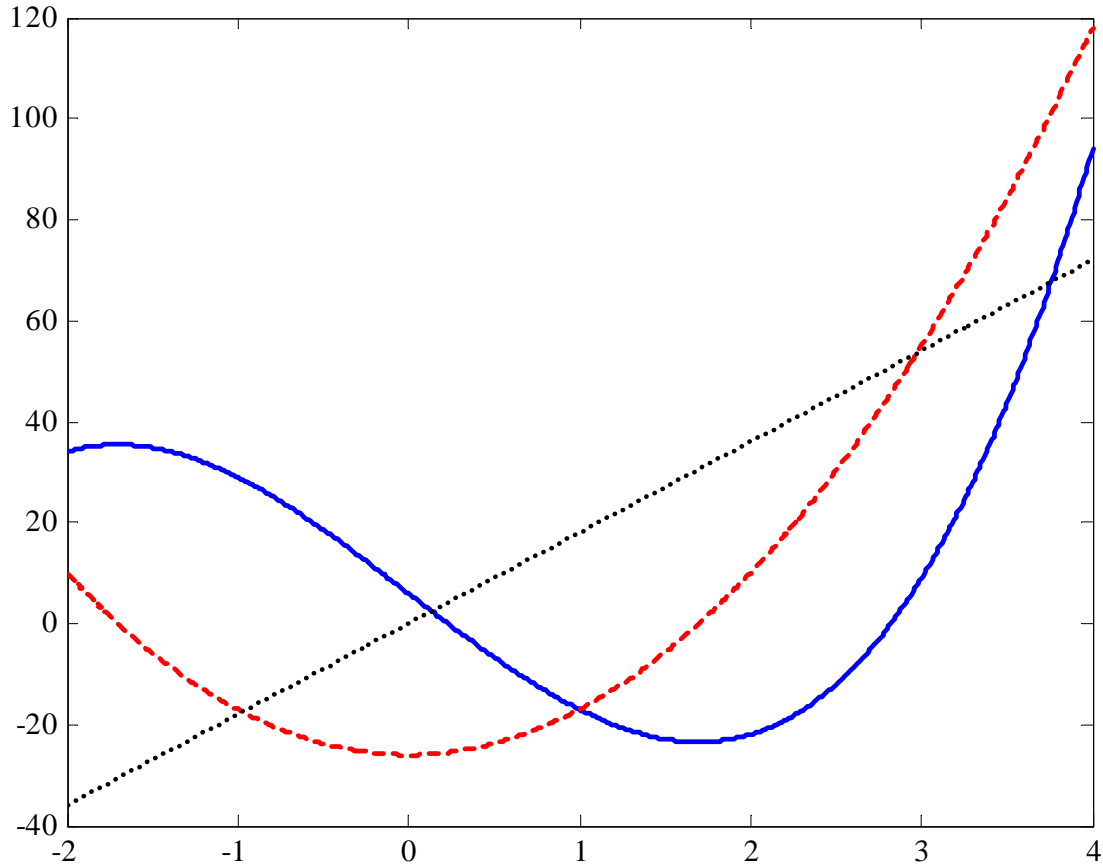
```
ydd = 18*x;
```

← Vector **ydd** with values of the second derivative.

```
plot(x,y,'-b',x,yd,'--r',x,ydd,':k')
```

← Create three graphs, **y** vs. **x** (solid blue line), **yd** vs. **x** (dashed red line), and **ydd** vs. **x** (dotted black line) in the same figure.

USING THE PLOT COMMAND TO PLOT MULTIPLE GRAPHS IN THE SAME PLOT



USING THE `hold on`, `hold off`, COMMANDS TO PLOT MULTIPLE GRAPHS IN THE SAME PLOT

`hold on` Holds the current plot and all axis properties so that subsequent plot commands add to the existing plot.

`hold off` Returns to the default mode whereby plot commands erase the previous plots and reset all axis properties before drawing new plots.

This method is useful when all the information (vectors) used for the plotting is not available at the same time.

USING THE hold on, hold off, COMMANDS TO PLOT MULTIPLE GRAPHS IN THE SAME PLOT

Plot of the function, $y = 3x^3 - 26x + 10$ and its first and second derivatives, for $-2 \leq x \leq 4$ all in the same plot.

```
x = [-2:0.01:4];  
y = 3*x.^3-26*x+6;  
yd = 9*x.^2-26;  
ydd = 18*x;  
plot(x,y,'-b')  
hold on  
plot(x,yd,'--r')  
plot(x,ydd,':k')  
hold off
```

← First graph is created.

← Two more graphs are created.

MATLAB ASSIGNMENT 5:

1. MATLAB book, Chapter 5, Problem 1.
2. MATLAB book, Chapter 5, Problem 4.
3. MATLAB book, Chapter 5, Problem 6.
4. MATLAB book, Chapter 5, Problem 7.

For each problem write a script file and execute it in the Command Window.

In each problem submit the script file and a print of the plot.