SEEE1022 INTRODUCTION TO SCIENTIFIC PROGRAMMING

CH7
Graphic

Dr. Mohd Saiful Azimi Mahmud (azimi@utm.my)
P19a-04-03-30, School of Electrical Engineering, UTM

OBJECTIVES

- 2-dimensional line plot function and the properties
- Sub-plot function
- Specialized 2-D plot functions
  1. Discrete graph: stem
  2. Log
  3. Histogram
  4. Polar
  5. Bar graph and pie chart

- Figure editing using wizard

INTRODUCTION

INTRODUCTION

WHY PLOTTING?

- Pictorial way of representing relationships between various quantities. It shows how one quantity changes if another quantity that is related to it also changes.
- Graph summarize huge information into one picture, thus easier to analyze the collected data.
- Easy way of comparing multiple measured data.

INTRODUCTION

SCIENTIFIC DATA

- In scientific data, each recorded value is called a sample. The collection of the samples form a data as a vector.
- Practically, each of this sample is recorded according to the change of another parameter such as time, angle, distance and etc. In math, this parameter is called function variable.
- For example, V(t) is a voltage reading over time, or voltage function relating to time.
- Thus, recording scientific data will at least save two types of vectors:
  1. The measured variable itself.
  2. The function variable.

INTRODUCTION

SCIENTIFIC DATA

No	Data	Measured Variable	Function Variable	No. of samples
1	Temperature recorded every $0.5s$ for 1 minute.	Temperature	Time. `0:0.5:60`	121
2	Voltage output of an RC circuit for 20 value of frequency range between $10Hz$ and $100Hz$.	Output Voltage	Frequency `linspace(10,100,20)`	20
3	Angle of a servo motor when supplied with 10 different input voltage values between $0V$ to $5V$	Angle	Input voltage `linspace(0,5,10)`	10
4	Electrocardiography (volt) reading of a heart captured at 200 sample per second for 1 minute	Volt	Time `0:1/200:60`	12001
5	Number of chairs in Classroom 1 until Classroom 10.	Total chairs	classroom	10

INTRODUCTION

PLOTTING DATA

- Plotting the scientific data will be as follows:
  1) y-axis – Measured variable
  2) x-axis – Function variable

- Both main variable and function variable vectors must have the same length.

2-D PLOT

2-D PLOT

2-D LINE PLOT FUNCTION

- Function `plot()` can be used to perform 2-D line plot.
- Syntax:

```
plot(X, Y, LineSpec)
```

- Description:
  1. Plot data in `Y` versus the corresponding values in `X`, which can be either scalar, vector or matrix. The size of variable `Y` and `X` must agreed.
  2. `X` data is optional. If not specified, `Y` data sample-number will be used.
  3. `LineSpec` is optional. This define the line style, marker symbol and colour of the plotted line.

2-D PLOT

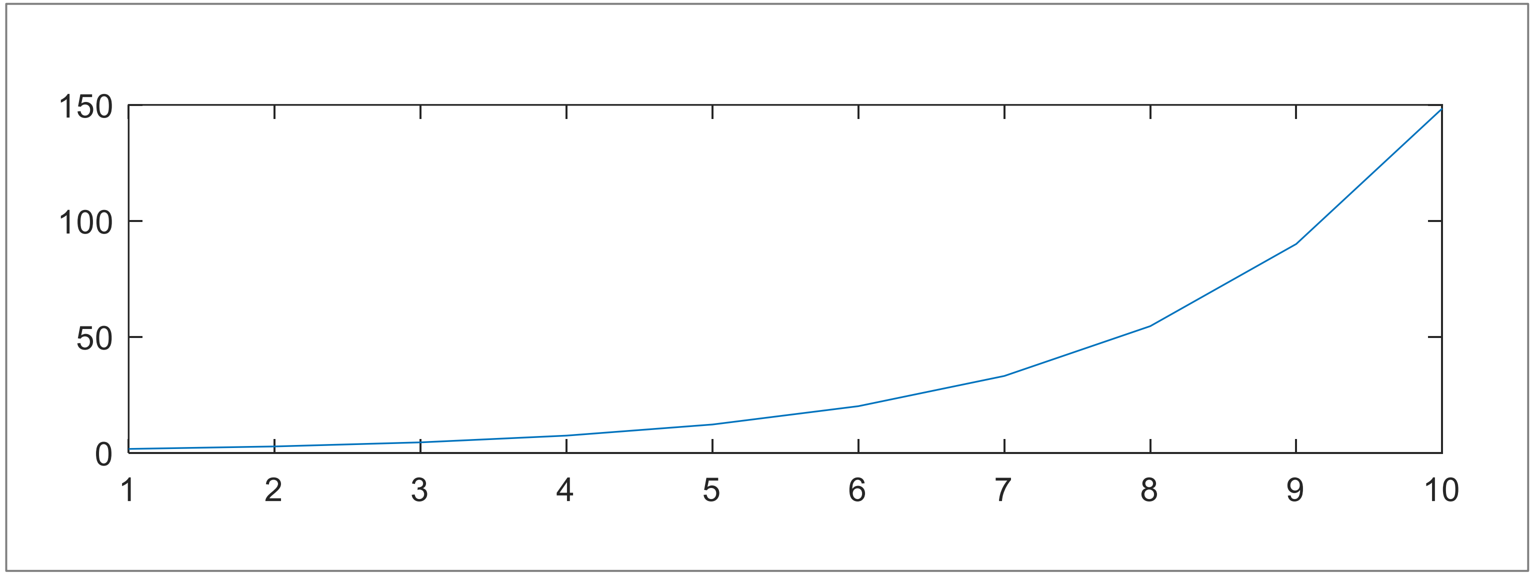
2-D LINE PLOT EXAMPLE

EXAMPLE 1

Plot a function $y = e^{x/2}$ for $x = [1,2,\ldots,10]$

```
x = 1:10;
y = exp(x/2);

plot(x,y);
```

TRY: `plot(y)`
What do you get and why?



2-D PLOT

Linespec

Line Style	Description
‘-’	Solid line (default)
‘—’	Dashed line
‘:’	Dotted line
‘-.’	Dash-dot line

Colour	Description
r	Red
g	Green
b	Blue
c	Cyan
m	Magenta
y	Yellow
k	Black
w	White

Marker	Description
+	Plus sign
o	Circle
*	Asterisk
.	Point
x	Cross
s	Square
d	Diamond
p	Pentagram
h	Hexagram
^	Upward-pointing triangle
v	Downward-pointing triangle
>	Right-pointing triangle
<	Left-pointing triangle

2-D PLOT

2-D LINE PLOT EXAMPLE

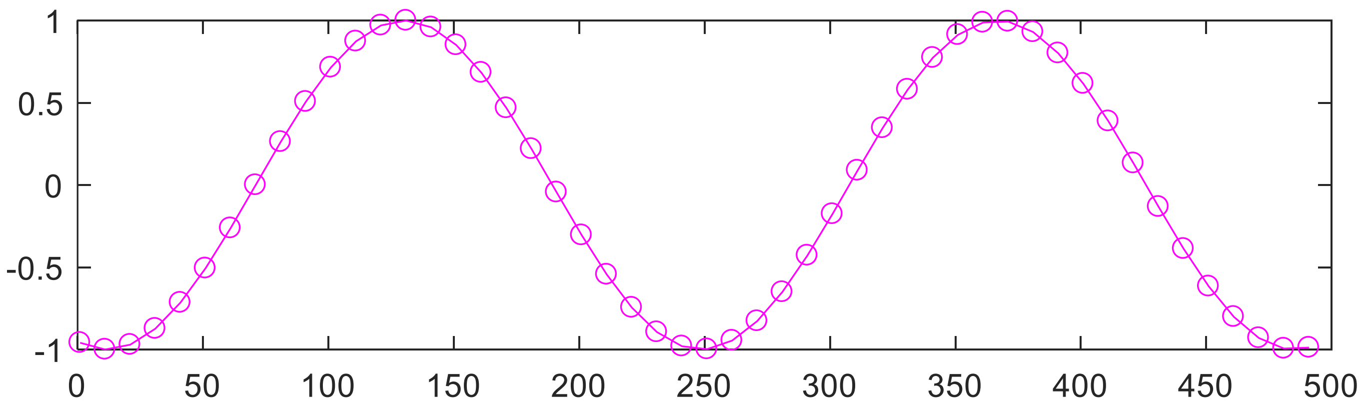
EXAMPLE 2

Plot a function $y = \sin(5x)$ for $x = \{1, 10, \dots, 500\}$

```
x = 1:10:500;
y = sin(5*x);

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

This time, `plot(y)` will not work. WHY?

`'- m o'` are `LineSpec` properties for solid line, magenta colour, and circle marker. Space between properties is optional.



2-D PLOT

PLOT FORMATTING

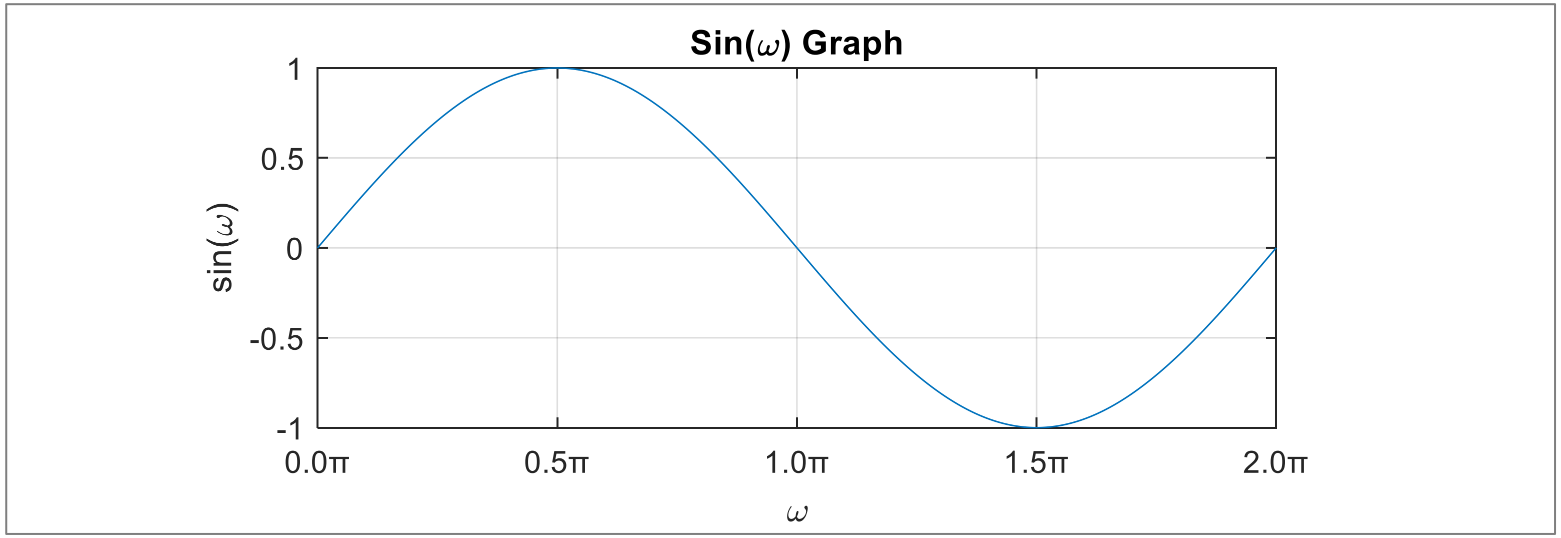
No	Properties	Function
1	Title and Label	`title`
		`xlabel, ylabel`
		`legend`
		`text`
2	Axes limits and aspect ratio	`xlim, ylim`
		`Axis`
3	Grid lines and tick	`grid`
		`xticks, yticks`
		`xticklabels, yticklabels`
		`xtickformat, ytickformat`
4	Multiple plots	`figure`
		`subplot`

2-D PLOT

2-D PLOT WITH SEVERAL PLOT PROPERTIES

EXAMPLE 2

```
w = linspace(0,2*pi,1000);
y = sin(w);
plot(w/pi,y)
xlabel('\omega'), ylabel('sin(\omega)')
xtickformat('%.2f\x3C0')
title('Sin(\omega) Graph')
grid on
```



2-D PLOT

MULTIPLE PLOT ON A SINGLE GRAPH

- Multiple plot drawn on the same graph is very useful when we want to compare the data with different settings.

- Syntax:

```
plot(X1, Y1, LineSpec1, X2, Y2, LineSpec2, …, Xn,
Yn, LineSpecn)
```

- For multiple plot, legend is use as the identifier for each plot. It can be set by using the following function:

```
legend ('plot1 legend','plot2 legend'…, 'plotn
legend')
```

2-D PLOT

MULTIPLE PLOT ON A SINGLE GRAPH

EXAMPLE 4

Visualize the intersection of the two polynomials below:

$$f(x) = 3x^4 + 2x^3 + 7x^2 + 2x + 9$$
$$g(x) = 5x^3 + 9x + 2$$

Plot the graph for $x = [-10, -9.5, -8.0, \ldots, 8.0, 9.5, 10]$

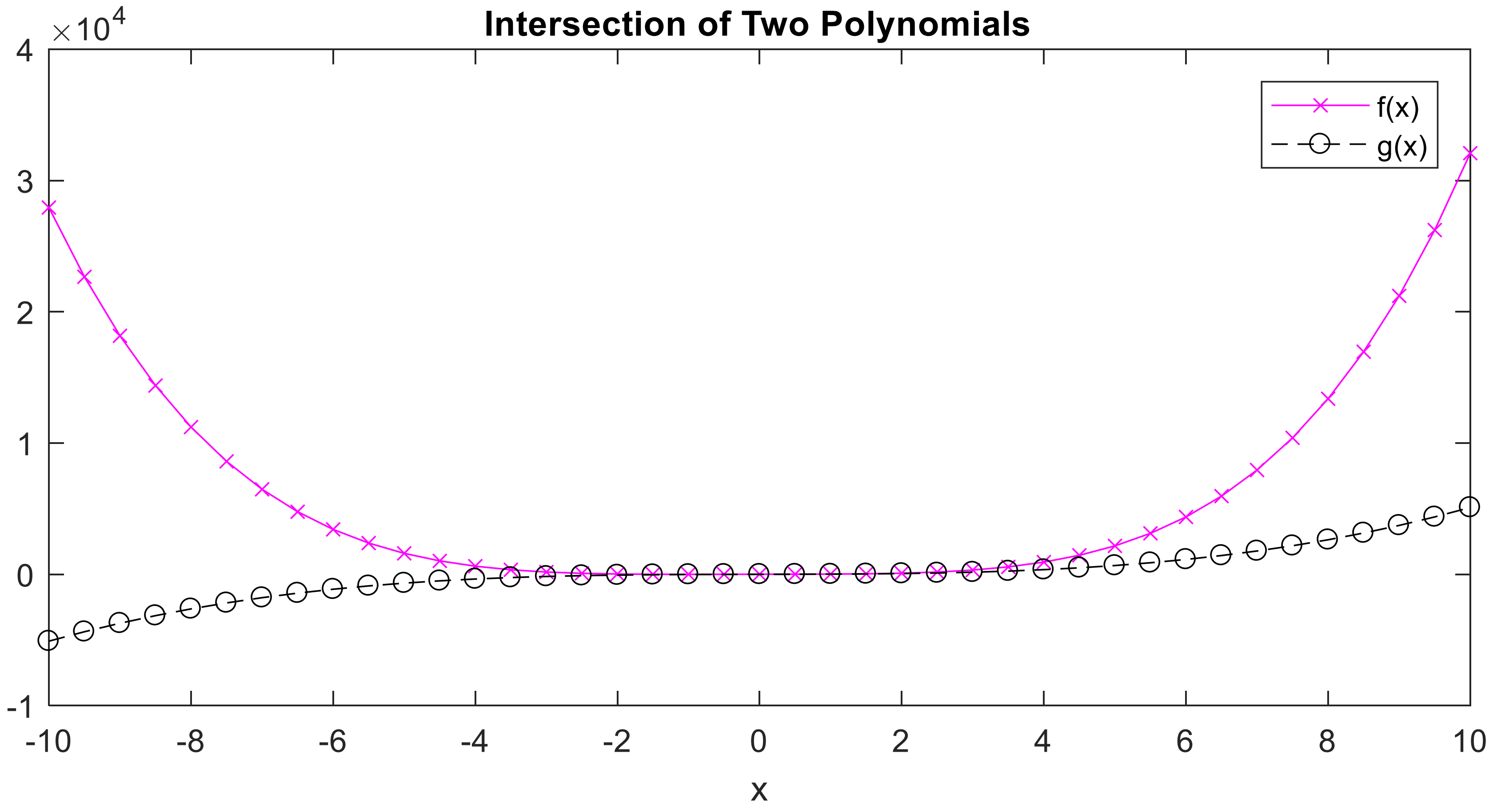
```
x = -10:0.5:10;
f = 3*x.^4 + 2*x.^3 + 7*x + 9;
g = 5*x.^3 + 9*x + 2;

plot(x,f,'- x m',x,g,'-- o k')
legend('f(x)','g(x)')
xlabel('x')
title('Intersection of Two Polynomials')
```

RECAP: `LineSpec` is optional, if it is not defined for any line plot, the default will be used instead

2-D PLOT

MULTIPLE PLOT ON A SINGLE GRAPH

EXAMPLE 4



2-D PLOT

SUBPLOT FUNCTION

- An array of plots can be created in the same figure, each of this plot is plotted as a **subplot**.
- **Syntax**:

```
subplot(m,n,p)
```

- Description:

1. `m` and `n`, specify the grid division size/slot on the current figure.
2. `p` define the position of the particular plot in the divided figure's grid. Column first then row (opposite to array linear indexing).

2-D PLOT

SUBPLOT : UPPER & LOWER PLOT

- Upper and lower subplot can be defied by 2-by-1 grid

EXAMPLE 5

Create a figure with two stacked subplots of y=cos(3x) and z=sin(x) for x=[1,2,…,10].

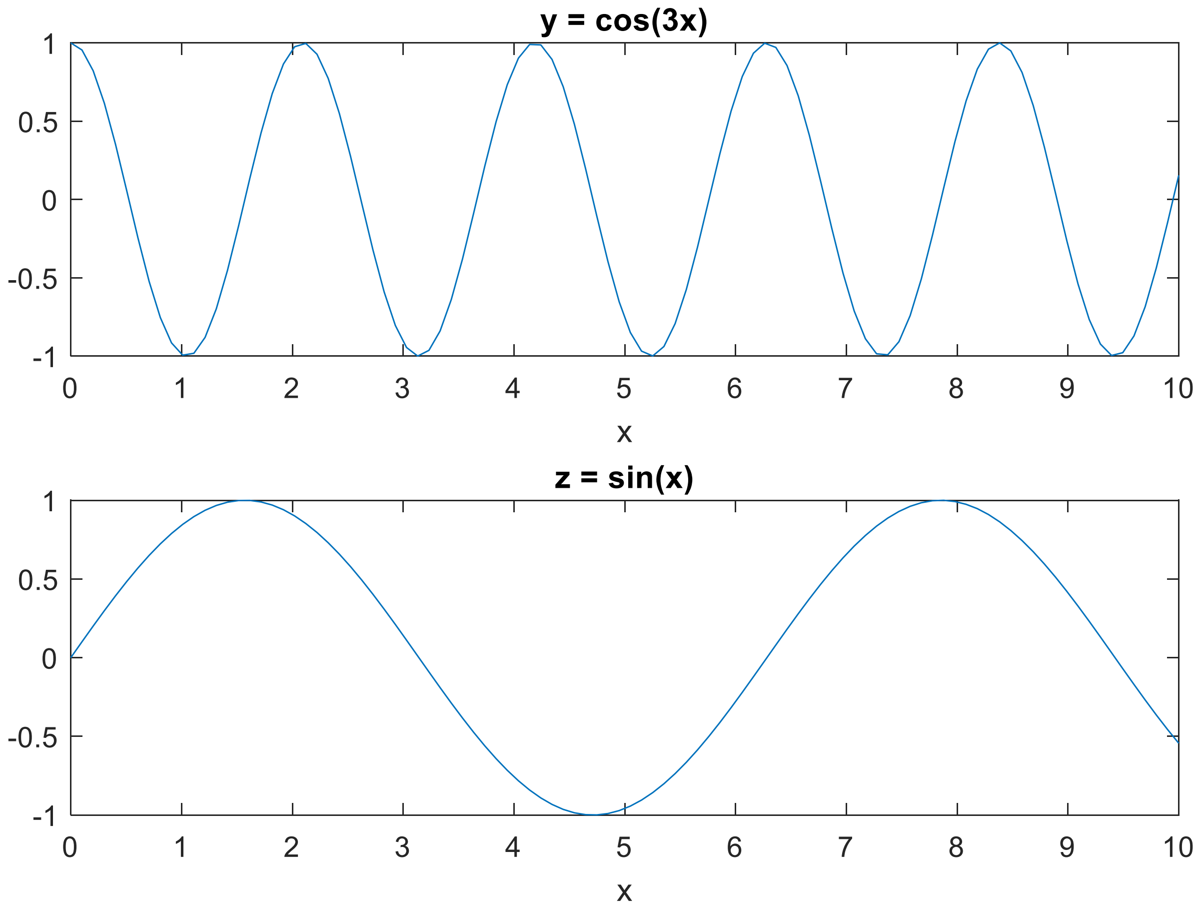
```
x = linspace(0,10,100);
y = cos(3*x);
z = sin(x);

subplot(2,1,1), plot(x,y)
title('y = cos(3x)');

subplot(2,1,2), plot(x,z)
tittle('z = sin(x)');
```

2-D PLOT

SUBPLOT : UPPER & LOWER PLOT

- Upper and lower subplot can be defied by 2-by-1 grid

EXAMPLE 5



2-D PLOT

SUBPLOT : RC CIRCUIT

EXAMPLE 6

- In analysing an RC circuit with an AC input voltage, two responses need to be plotted. They are magnitude and phase responses. Below is the equation to obtain both responses where R is resistor value in ohm, C is the capacitor value in farad and f is the frequency of the AC input voltage:

$$H = \frac{1}{1 + j2\pi fRC}$$

$$Magnitude = \frac{1}{\sqrt{1+(2\pi fRC)^2}} \quad , \qquad Phase = -\tan^{-1}(2\pi fRC)$$

- In MATLAB, we can use function `abs(H)` and `angle(H)` to automatically compute $Magnitude$ and $Phase$ from H respectively.
- $Magnitude$ is a ratio, thus no unit. On the other hand, the unit of $Phase$ is radian.
- Write a MATLAB script to plot both of the $Magnitude$ and $Phase$ responses when $R = 4.7\text{k}\Omega$ and $C = 0.47\mu F$ for the value of f between $0Hz$ to $10kHz$.

2-D PLOT

SUBPLOT : RC CIRCUIT

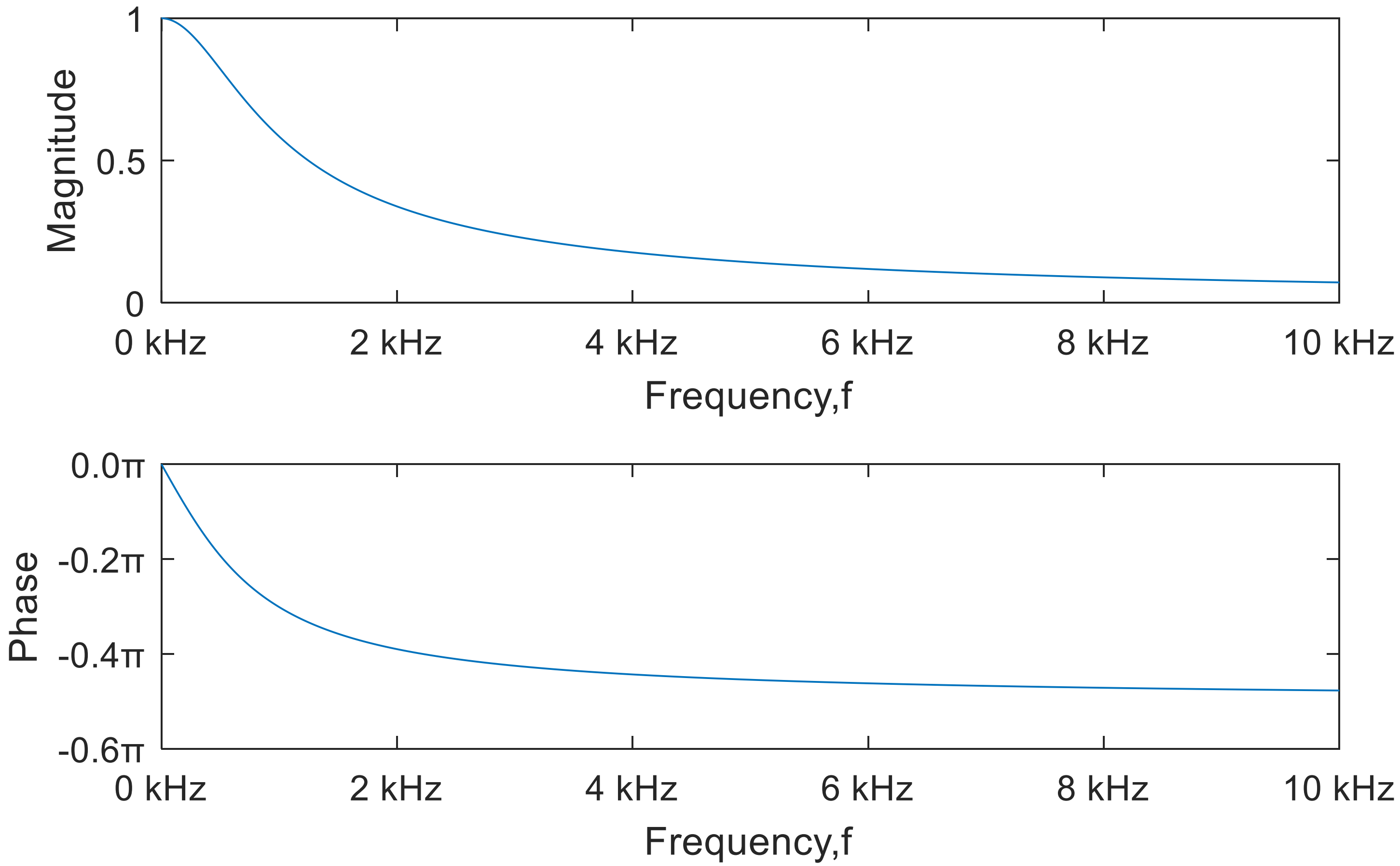
EXAMPLE 6

```
C = 0.47e-6;
R = 4.7e3;
f = 0:10000;
H = 1./(1+1i*2*pi*f*R*C);
Magnitude = abs(H);
Phase = angle(H);

subplot(2,1,1), plot(f/1000,Magnitude)
xlabel('Frequency,f')
xtickformat('%.0f kHz')
ylabel('Magnitude’)

subplot(2,1,2), plot(f/1000,Phase/pi)
xlabel('Frequency,f')
xtickformat('%.0f kHz')
ylabel('Phase')
ytickformat('%.1f\x3C0')
```

2-D PLOT

SUBPLOT : RC CIRCUIT

EXAMPLE 6



SUBPLOT : QUADRANT

- Quadrant subplot can be defied by 2-by-2 grid.

EXAMPLE 7

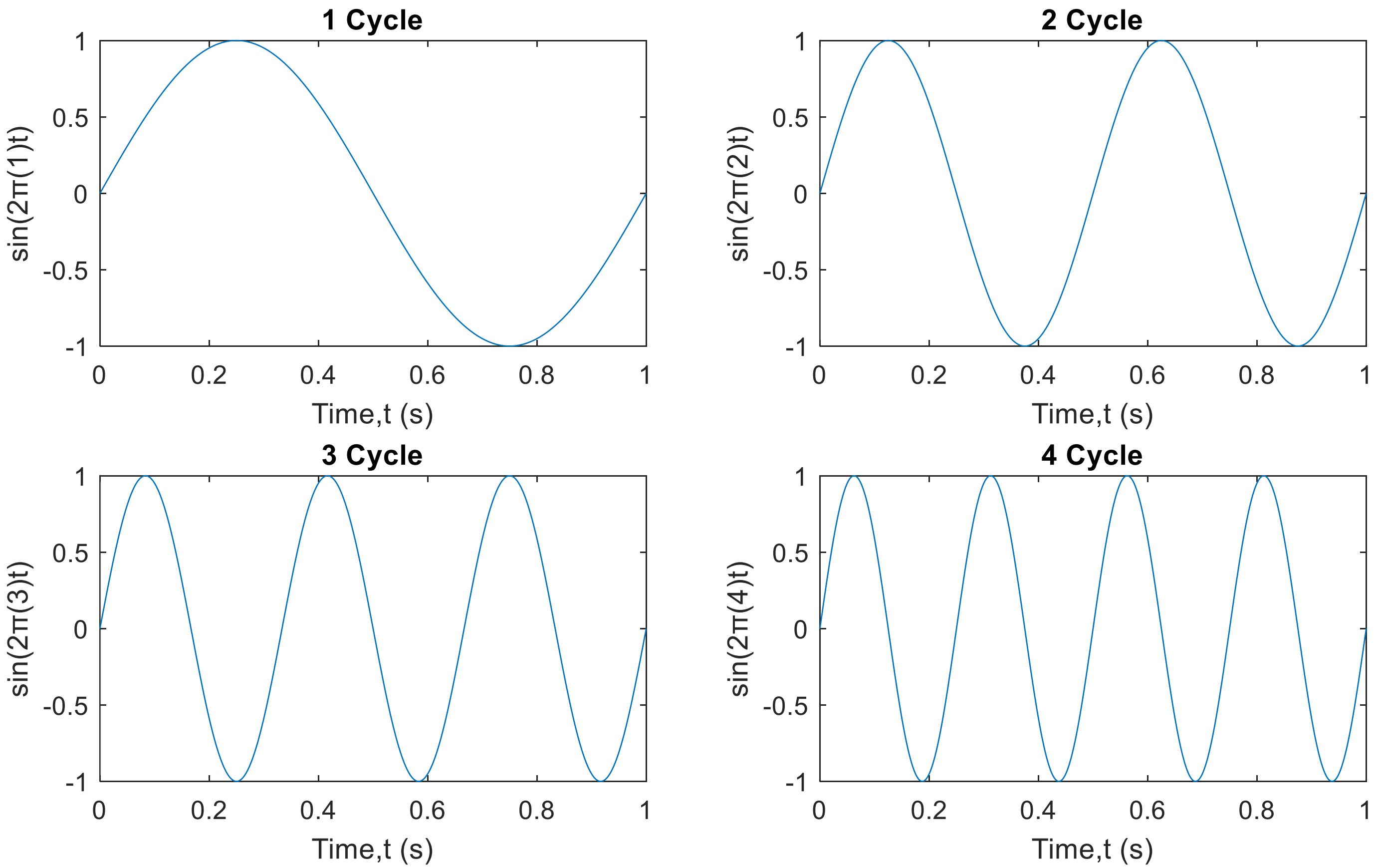
Plot 4 sinusoidal signals of 4 different frequencies ($F = 1Hz, 2Hz, 3Hz, 4Hz$) on a single figure. Plot the signals for $t = [0{:}\,0.001{:}\,1]$.

```
t = 0:0.001:1;
for F = 1:4
    y = sin(2*pi*F*t);
    subplot(2,2,F), plot(t,y)
    xlabel('Time,t (s)')
    ystring = sprintf('sin(2\x3C0(%d)t)',F);
    ylabel(ystring)
    titlestring = sprintf('%d Cycle',F);
    title(titlestring)
end
```

SUBPLOT : QUADRANT

- Quadrant subplot can be defied by 2-by-2 grid.

EXAMPLE 7



SPECIAL PLOT

DISCRETE GRAPH : STEM

- `stem()` function is used to plot discrete sequence data
- **Syntax:**

```
stem(X, Y, LineSpec)
```

Description:

1) Plot the data in `Y`, at value specified by `X`.
2) `X` data is optional. If not specified, `Y` data sample-number will be used.
3) `LineSpec` is optional. This define the line style, marker symbol and colour of the plotted line.

SPECIAL PLOT

STEM : DISCRETE SIGNAL

EXAMPLE 8

Plot discrete signal of $\sin(\omega n)$ for $n = [1,2,3, \dots, 100]$ by setting $\omega = 0.04\pi$.

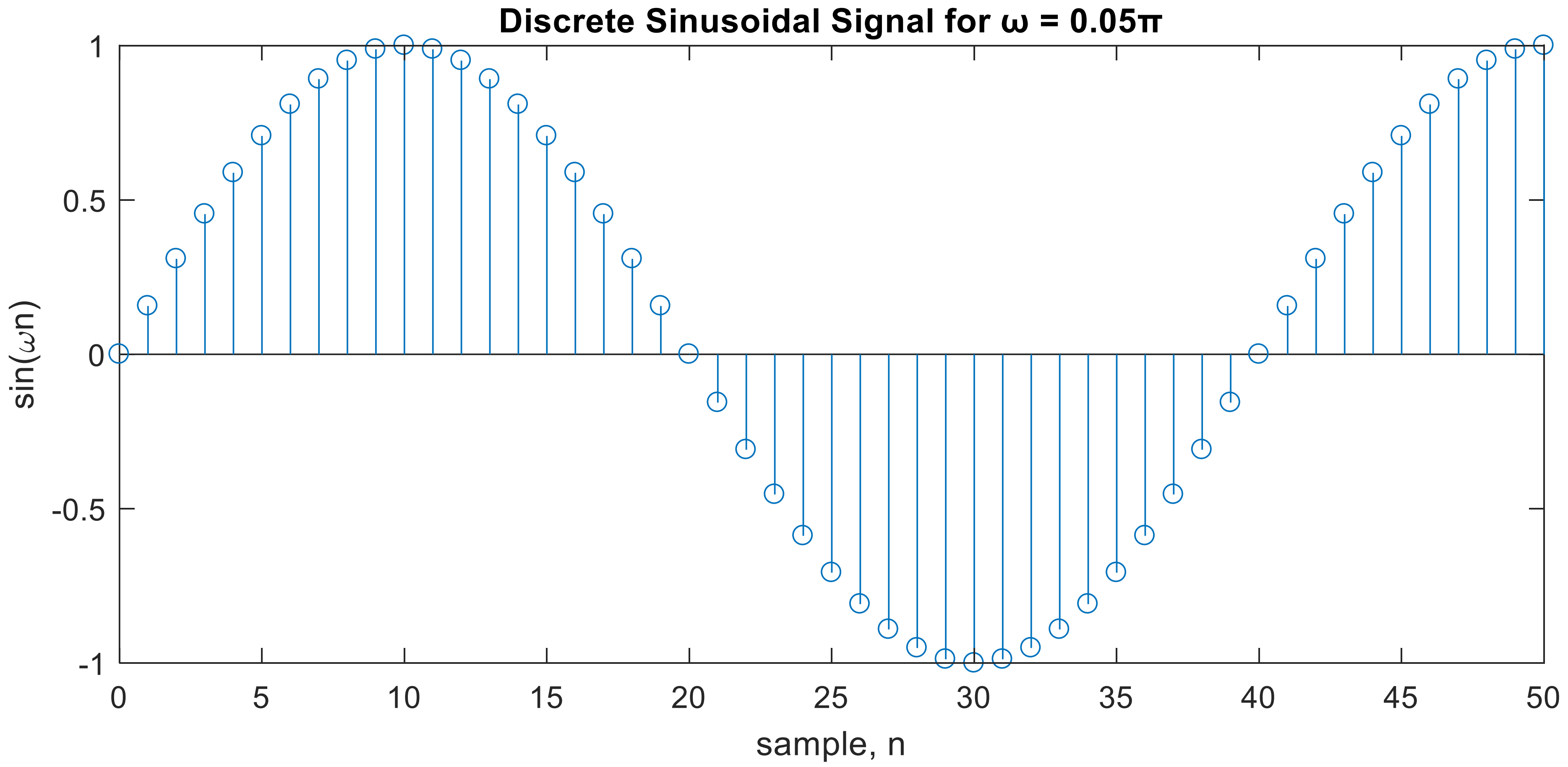
```
n = 0:100;
w = 0.04*pi;
y = sin(w*n);

stem(n,y)
xlabel('sample, n')
ylabel('sin(\omegan)')
t_string = sprintf(['Discrete Sinusoidal Signal' ...
                    ' for \x3C9 = %.2f\x3C0'],w/pi);
title(t_string)
```

Try change $\omega = 0.01\pi$ and observe the advantage of using function `sprintf`

SPECIAL PLOT

STEM : DISCRETE SIGNAL

EXAMPLE 8



SPECIAL PLOT

LOGARITHMIC PLOT

- A logarithmic scale is a nonlinear scale used when there is a large range of quantities.
- Common uses include:
  - Earthquake strength (Richter Magnitude).
  - Sound loudness (Decibel).
  - pH for acidity.
  - Frequency range (Bode plot).

- **Syntax:**

```
semilogx(X,Y,LineSpec)    %x-axis log plot

semilogy(X,Y,LineSpec)    %y-axis log plot

loglog(X,Y,LineSpec)      %both axis log plot
```

SPECIAL PLOT

LOUDNESS

EXAMPLE 9

Environment loudness (strength of sound) was recorded on several occasion as below. Plot the data using normal plot and logarithmic plot. Then, compare the two plots.

Occasion	Loudness (watt)
Library	7.94×10^{2}
Cafeteria	1.26×10^{6}
Inside travelling car	3.16×10^{8}
Train station	1.58×10^{9}
Riding motorcycle	6.3×10^{9}
Rock concert	1.99×10^{11}

SPECIAL PLOT

LOUDNESS

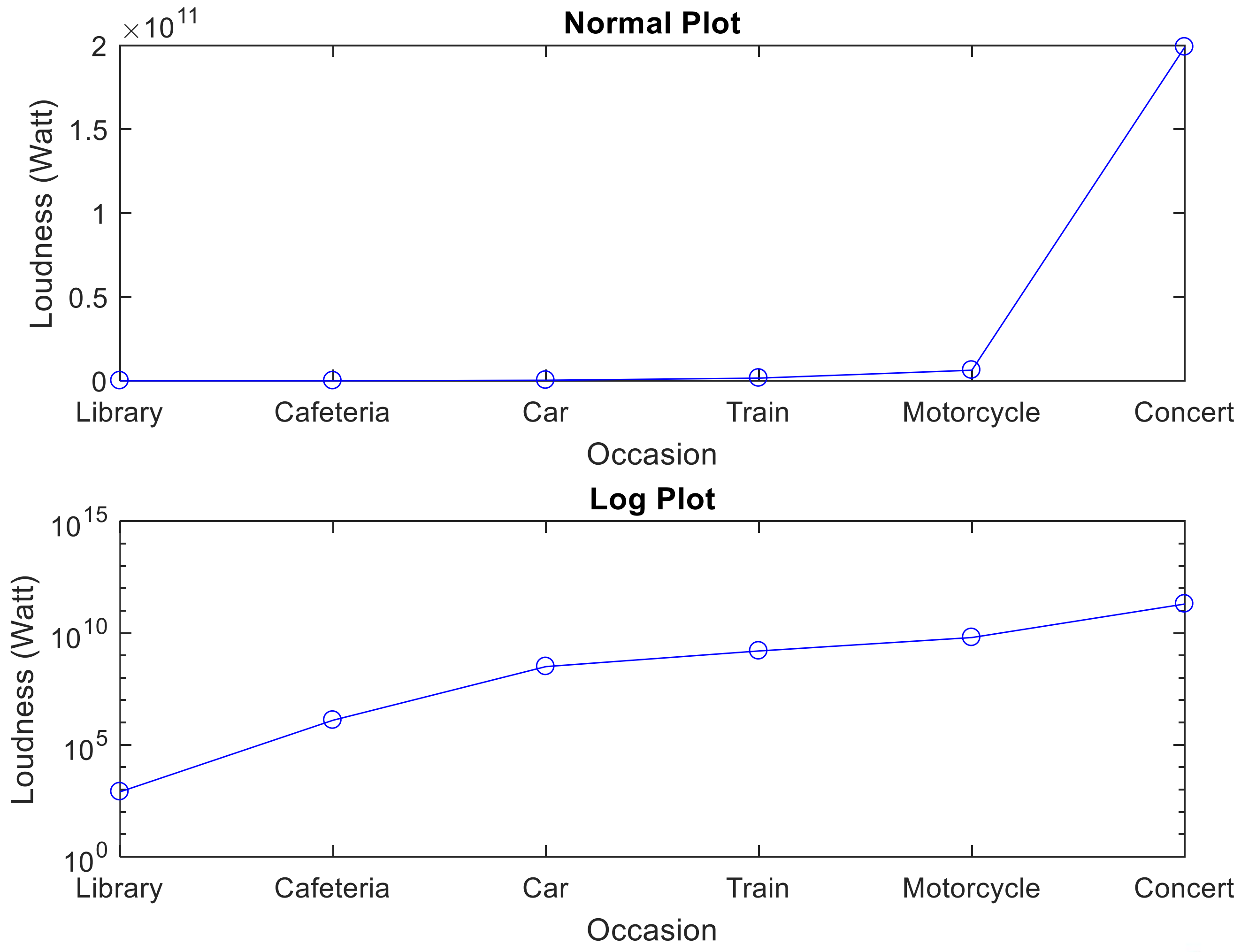
EXAMPLE 9

```
L = [7.94e2, 1.26e6, 3.16e8, 1.58e9, 6.3e9, 1.99e11];
Occasion = {'Library','Cafeteria','Car' ...
            'Train','Motorcycle','Concert'};

subplot(2,1,1), plot(L,'- b o')
xticks(1:6)
xticklabels(Occasion)
xlabel('Occasion')
ylabel('Loudness (Watt)')
title('Normal Plot')

subplot(2,1,2), semilogy(L,'- b o')
xticks(1:6)
xticklabels(Occasion)
xlabel('Occasion')
ylabel('Loudness (Watt)')
title('Log Plot')
```

SPECIAL PLOT

LOUDNESS

EXAMPLE 9



SPECIAL PLOT

BODE PLOT

EXAMPLE 10

Back to Example 6, $Magnitude$ and $Phase$ are actually better plotted using bode plot since there are important information in the lower frequency range. Below is how they are plotted with bode plot.

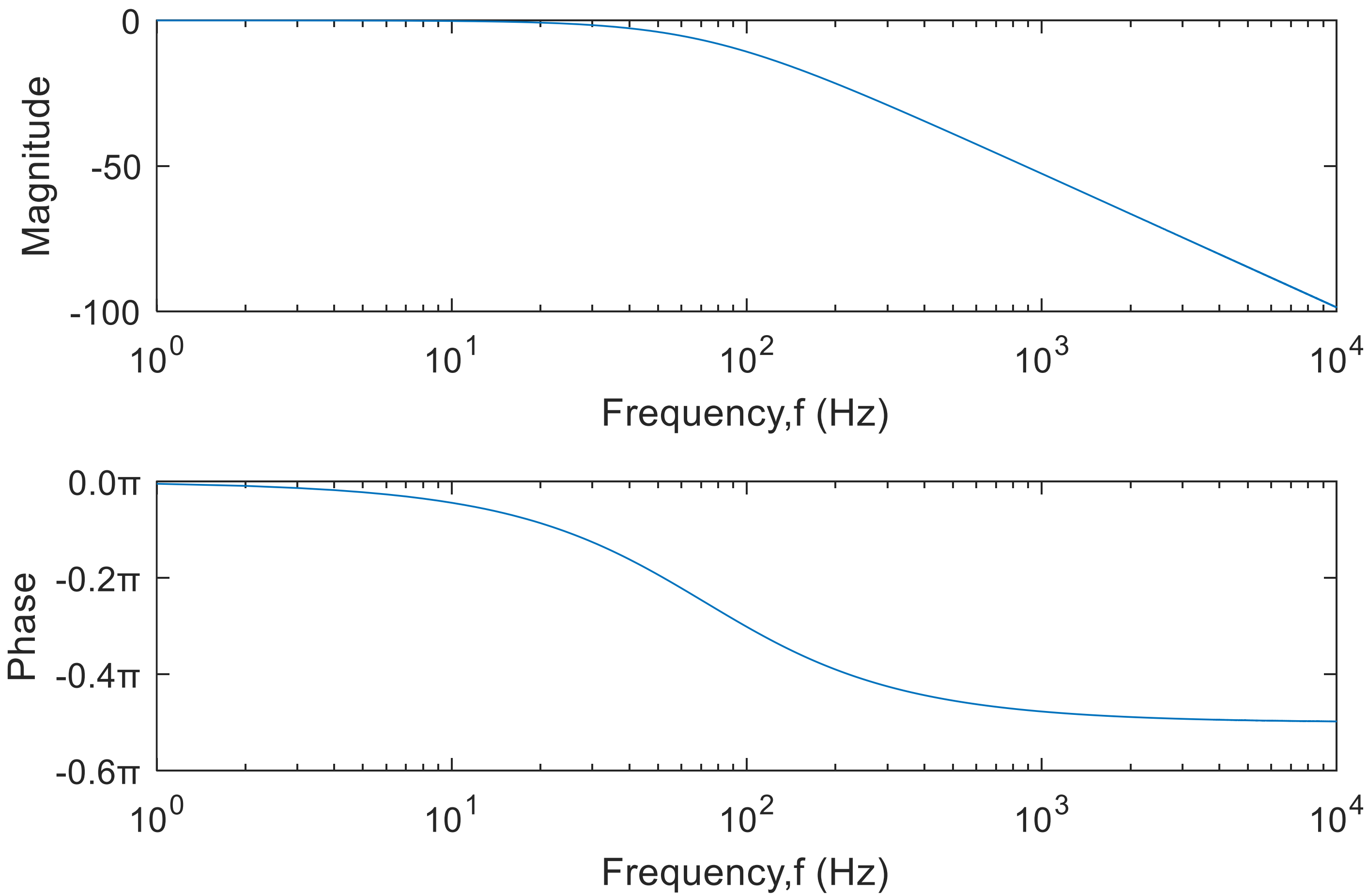
```
C = 0.47e-6; R = 4.7e3; f = 0:10000;
H = 1./(1+1i*2*pi*f*R*C);
Magnitude = 20*log(abs(H));
Phase = angle(H);

subplot(2,1,1), semilogx(f,Magnitude)
xlabel('Frequency,f (Hz)')
ylabel('Magnitude')
subplot(2,1,2), semilogx(f,Phase/pi)
xlabel('Frequency,f (Hz)')
ylabel('Phase')
ytickformat('%.1f\x3C0')
```

Note that $Magnitude$ is also better represented in log. However, it is already done via formulation. Thus no need log plot for the $Magnitude$ (y-axis)

SPECIAL PLOT

BODE PLOT

EXAMPLE 10



SPECIAL PLOT

HISTOGRAM

- Histogram plot could be performed by using `histogram()` function.
- The histogram function automatically chooses an appropriate number of bins with a uniform width to cover the range of values in the data list, and show the shape of the underlying distribution.
- **Syntax:**

```
histogram(X,nbins)
histogram(X,edges)
```

Description:

1) `X` is the data to be plotted on the histogram.
2) `nbins` a scalar defining the custom number of user defined bins.
3) `edges` is a vector defining the bin edges of the histogram.

SPECIAL PLOT

HISTOGRAM : TEST SCORE

EXAMPLE 11

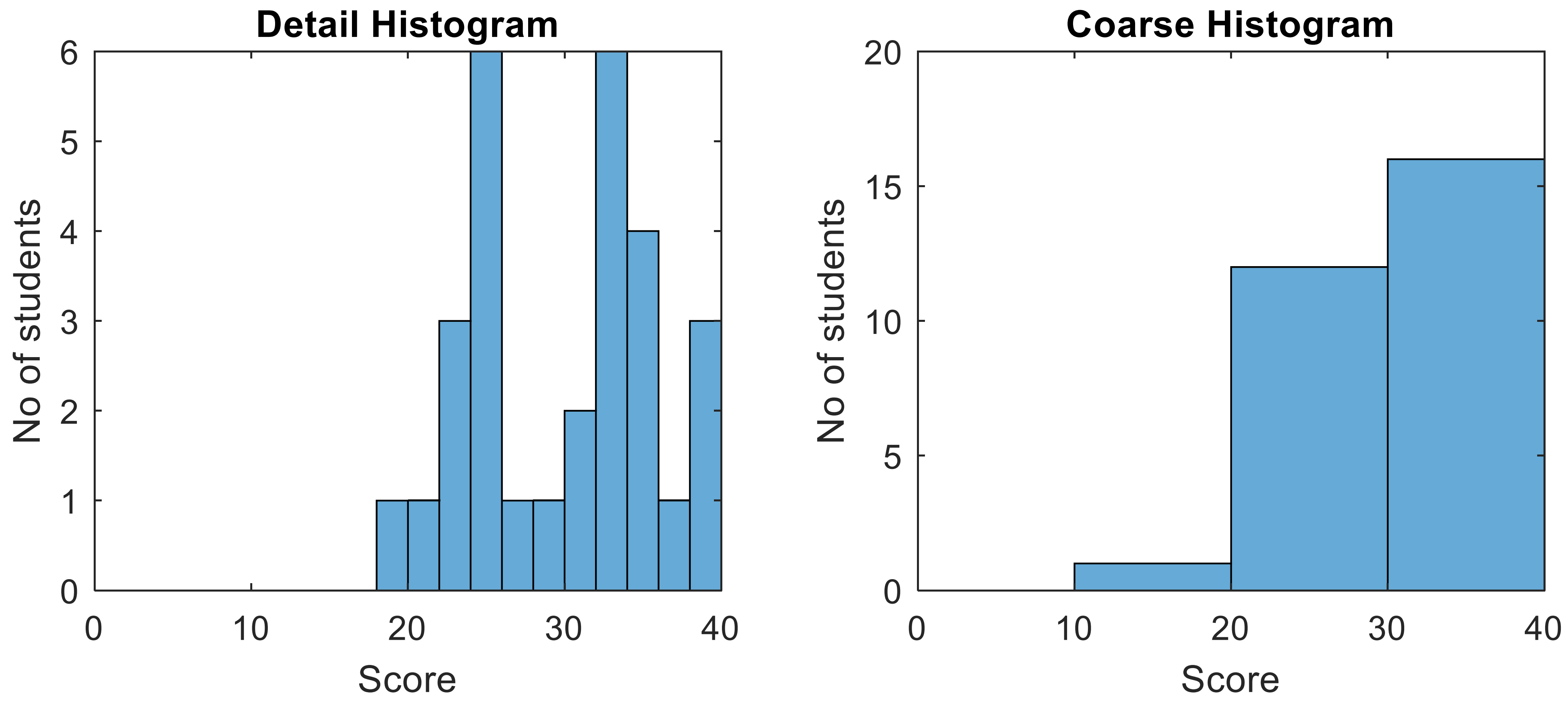
Below is a program code to visualize the test 1 scores of a class when uniformly group into 4 and 20 score ranges.

```
marks = xlsread(['MARKS-2017181-SKEE1022 - Sec01' ...
                 ' SKEL.xlsx'],2,'E6:E34');

subplot(1,2,1), histogram(marks,0:2:40)
xlabel('Score');
ylabel('No of students')
title('Detail Histogram')

subplot(1,2,2), histogram(marks,0:10:40)
xlabel('Score');
ylabel('No of students')
title('Coarse Histogram')
```

SPECIAL PLOT

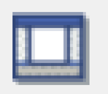
HISTOGRAM : TEST SCORE

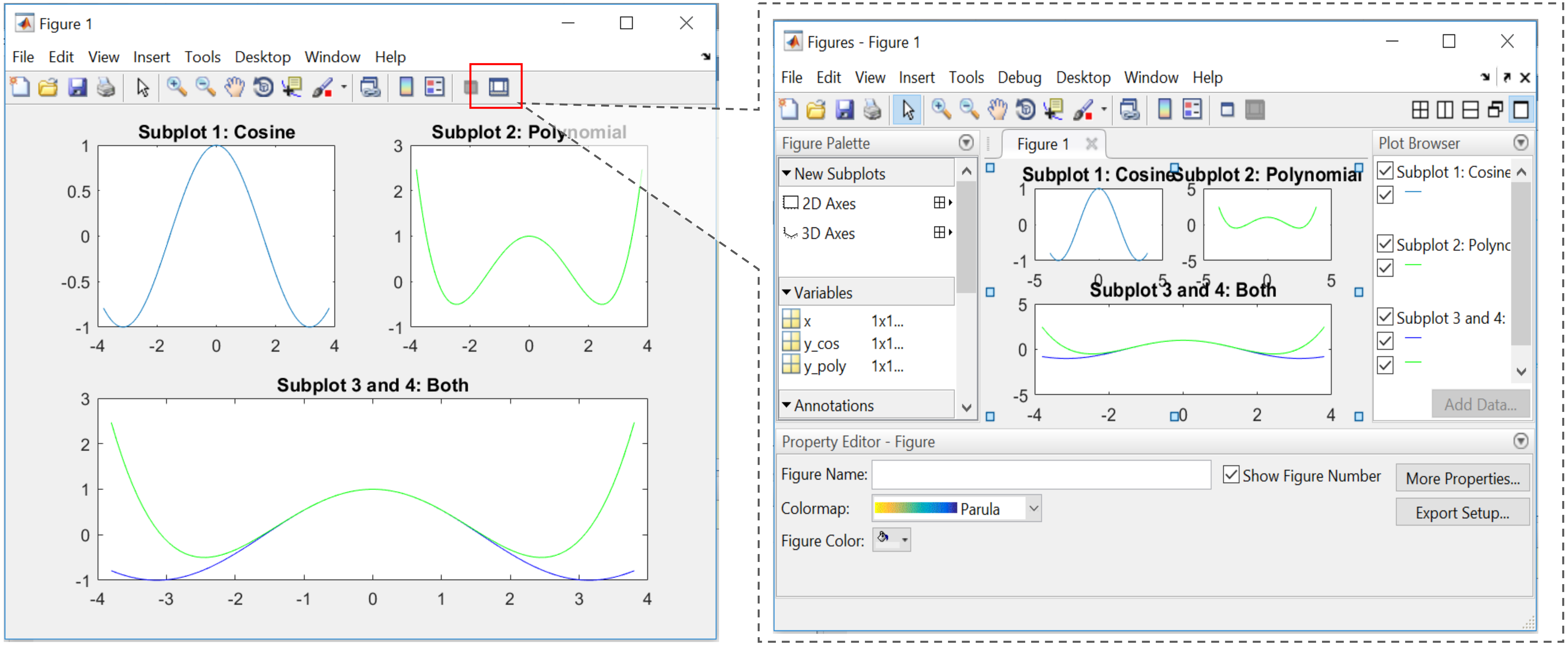
EXAMPLE 11



PLOT EDITING USING PLOT TOOLS

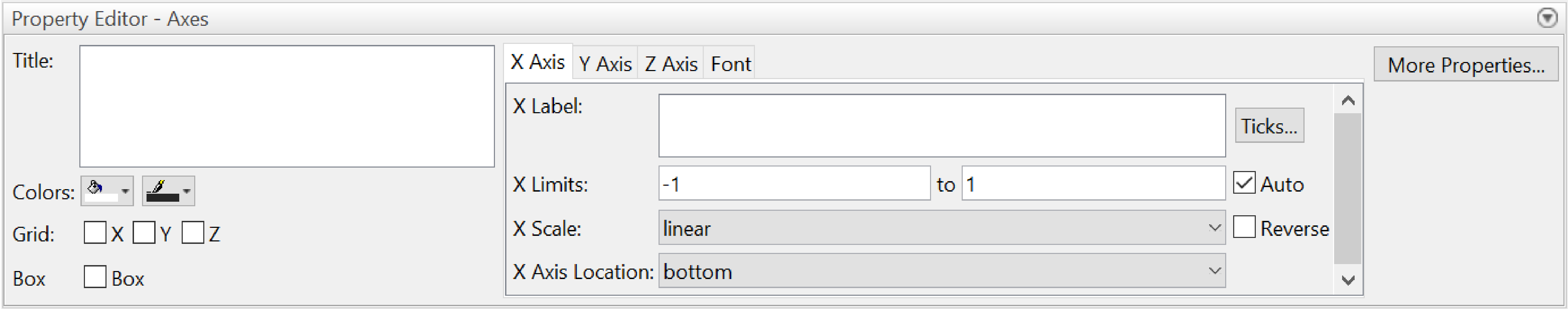
PLOT TOOLS

- Alternatively, MATLAB plot/figure could be customized using a GUI based interface called Plot Tools.
- Plot Tools icon  is available in the figure window.



PLOT TOOLS

- To customize objects in your graph, you can set their properties using the Property Editor. For example, click the axes to display a subset of common axes properties in the Property Editor. Specify a title and an x-axis label by typing text in the empty fields.



UTM
UNIVERSITI TEKNOLOGI MALAYSIA

univteknologimalaysia

utm_my

utmofficial

Thank You

www.utm.my

innovative ● entrepreneurial ● global