THE UNIVERSITY OF RHODE ISLAND

EGR 106
Foundations of Engineering II

Week 7 Lecture – Part B
2D Plotting

This Week's Topics

2D Plotting

- plot command
- line specifiers and properties
- multiple graphs
- formatting a plot
- logarithmic axes
- special graphs - histograms, pie charts, polar plots, etc.
- 3D plotting (brief intro)

2D Plotting

Graphical presentation has become the standard method to show technical information. Engineers use plots to analyze, visualize, and present their work.

Matlab provides many powerful plotting tools.

We'll review some of them today.

A Simple Example

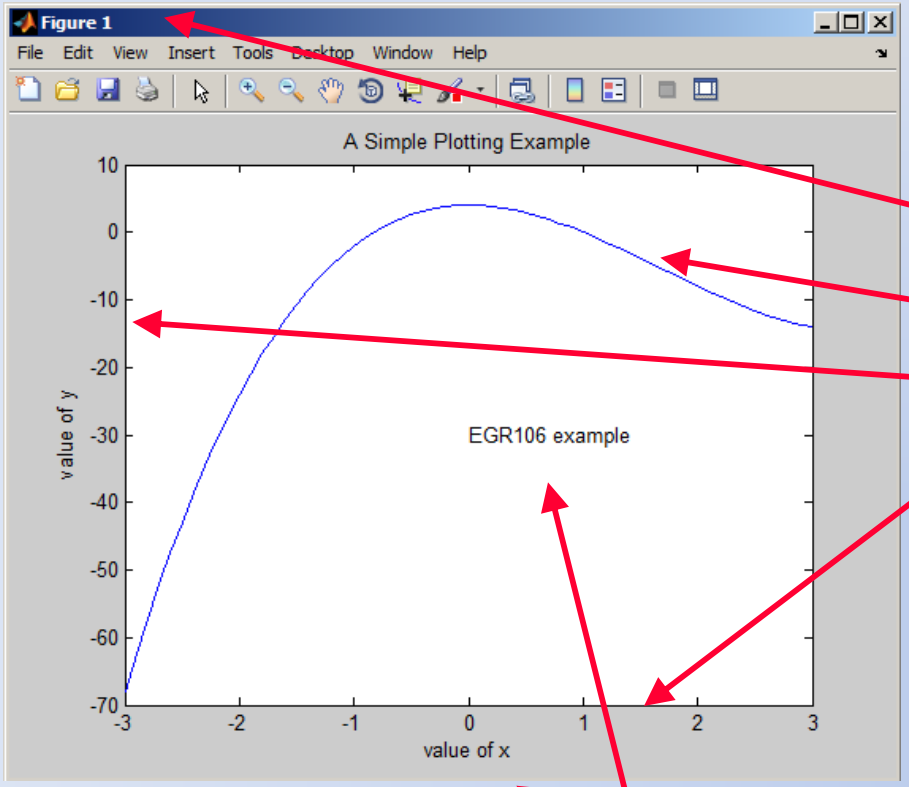
colon and dot notation for arrays

```
x =  –3:0.1:3;
y = x.^3 – 5*x.^2 + 4;
plot(x,y)
xlabel('value of x')
ylabel('value of y')
title('A Simple Plotting Example')
text(0,-30, 'EGR106 example')
```

standard form for plot

annotation tools

Resulting Plot



By default:

- Figure is “Figure 1”,
- Lines are blue
- Scales are automatic

note axis labels & text

Figure Formatting

Two Approaches:

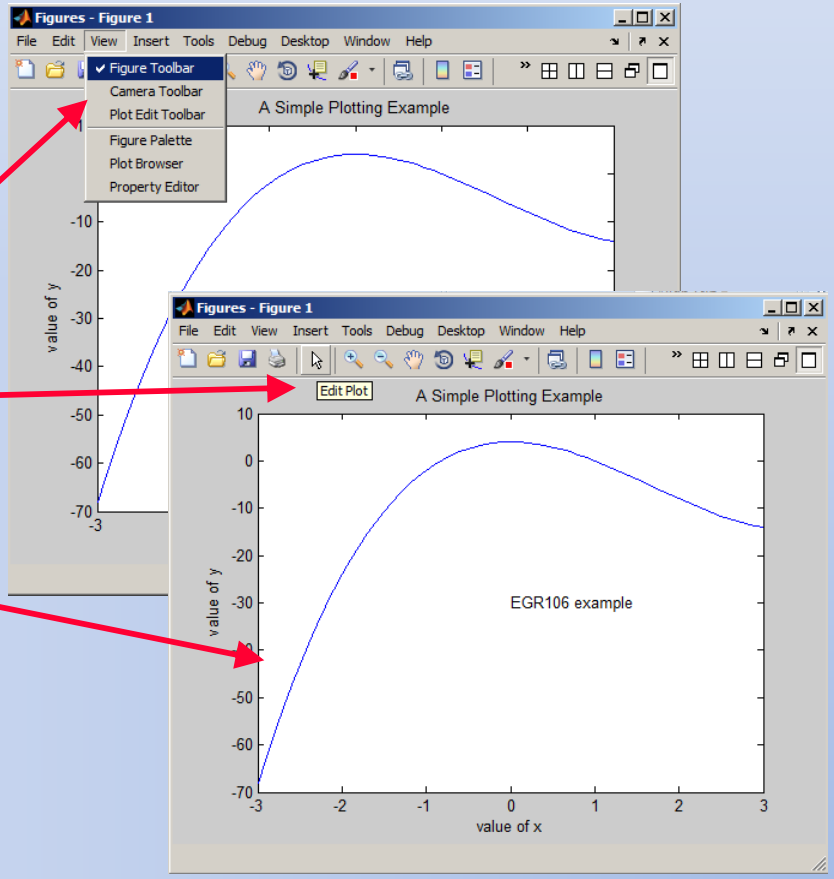
- Interactive figure editing
- Command line formatting (recommended)

Interactive Figure Editing

Select View => Figure Toolbar

Select “Edit Plot”

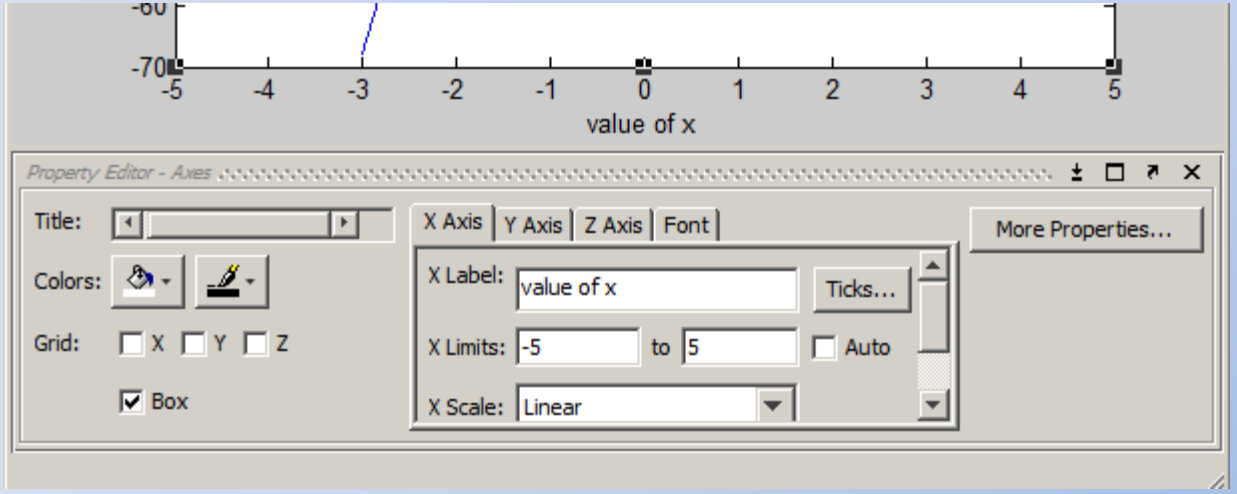
Double click on object you’d like to edit – opens “Property Editor”



Property Editor

Provides interactive editing of text, axes, line properties, marker properties

Example – axis editor:



Property Editor vs. Command Line Formatting

Property editor is like a spreadsheet's tools, but ...

- is limited to a single figure
- is tedious to repeat for other plots

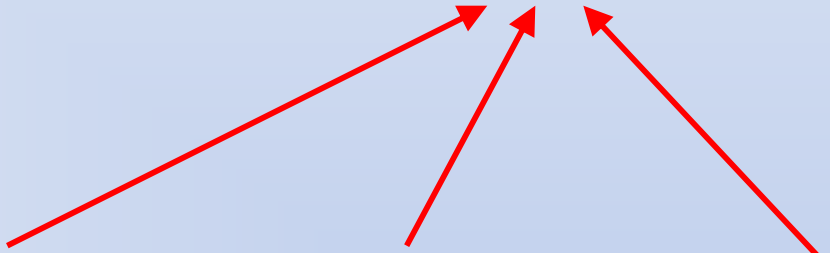
Often more efficient to use command line plot formatting:

plot(x,y, 'linespec' , 'Propname' , PropValue)

- line specifiers: color, line type, markers for data
- property name and value: thickness, size, etc

Line Specifiers

plot(x,y, ' ')



Color:
- k black
- r red
- b blue
- g green
- y yellow
- c cyan
- w white
- m magenta

Symbol (Marker):
- . point
- o circle
- x x-mark
- s square
- d diamond
- etc.

Line Style:
- - solid
- : dotted
- -- dashed
- -. dash-dot

Note: The order is not important !

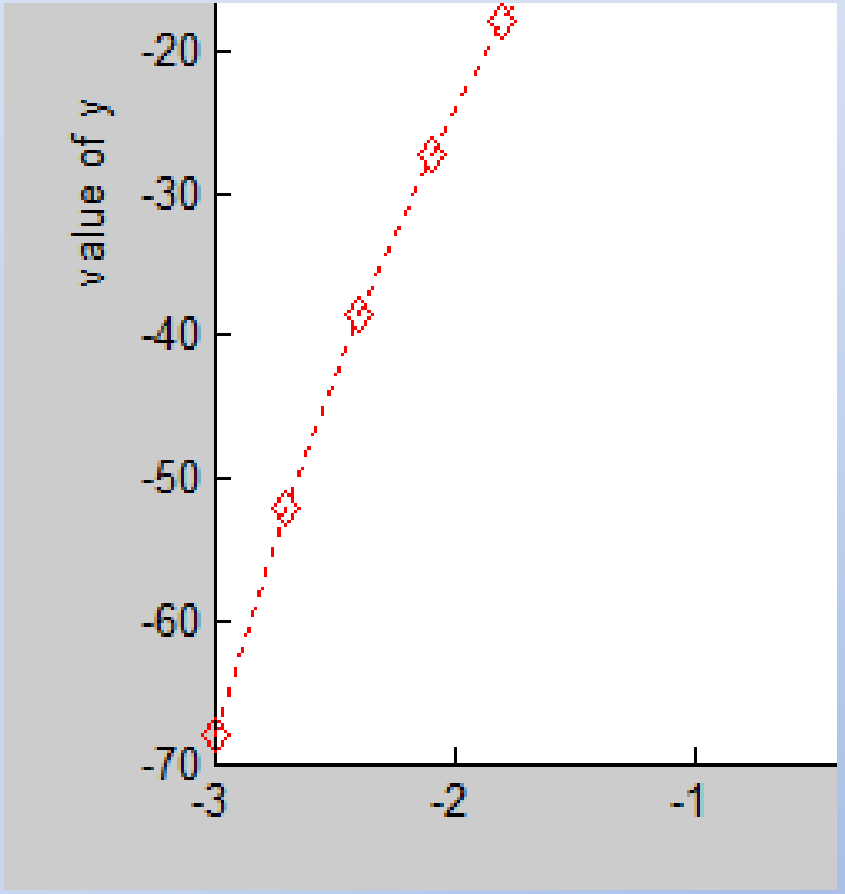
Line Specifiers - Example

plot(x,y, ' r : d ')

red

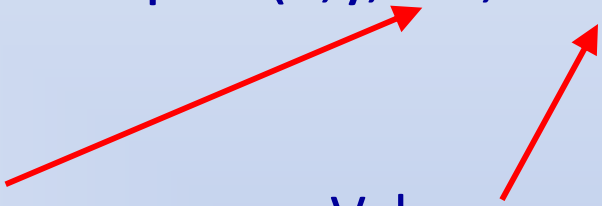
dotted line

diamonds



Line Properties

plot(x,y,’ ’,value)



Properties:

- linewidth
- markersize
- markeredgecolor
- markerfacecolor

Value:

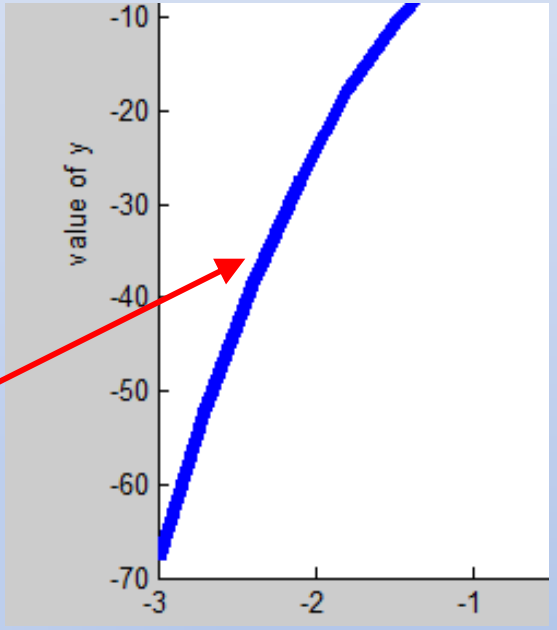
- varies with each property
- sizes in points
- colors as strings

Note: Can have multiple properties in one command

Line Properties - Example

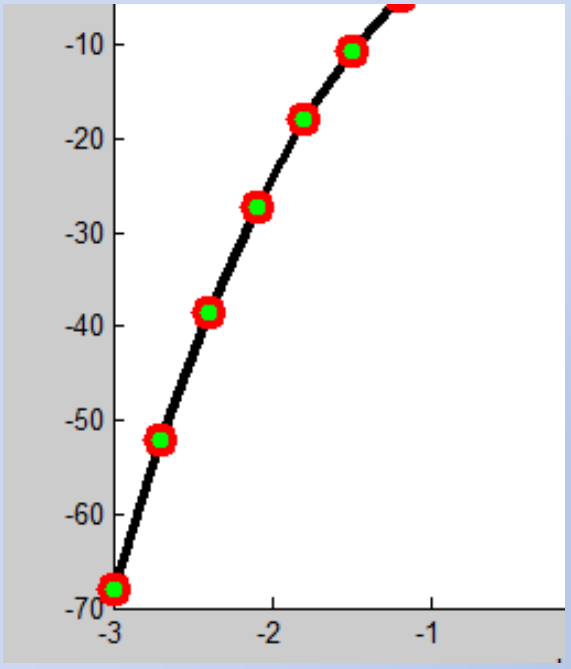
plot(x,y, 'linewidth',5)

line width is 5 “points”



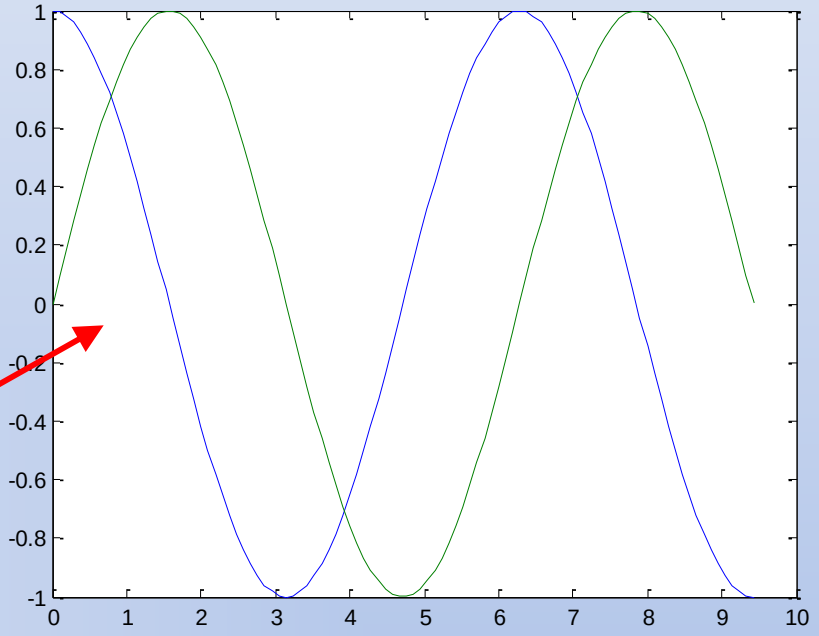
Line Properties – Another Example

```
plot(x,y, '- k o' , 'LineWidth' , 3, 'MarkerSize', 10,…
        'MarkerEdgeColor','red','MarkerFaceColor','green')
```



Multiple Graphs on Same Plot

```
x=linspace(0,3*pi);
y1=cos(x);
y2=sin(x);
plot(x,y1,x,y2)
```

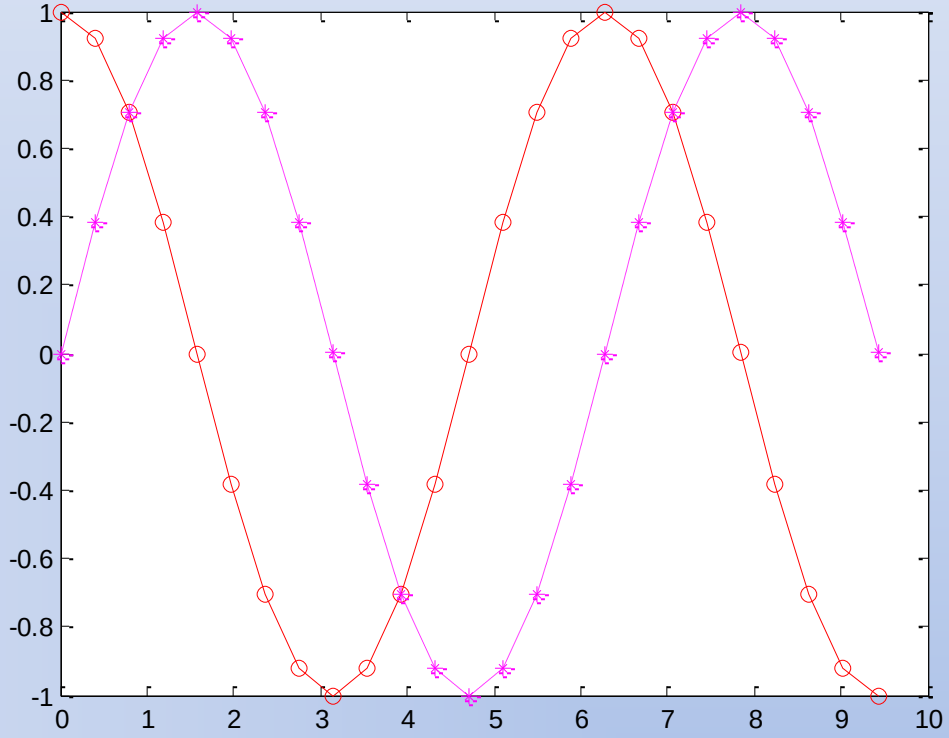


Note: Default is solid line with ‘rotating’ colors

Multiple Graphs (cont.)

```
x=linspace(0,3*pi,25);
y1=cos(x);
y2=sin(x);
plot(x,y1,'ro-',x,y2,'m*--')
```

Each line can have its own specification



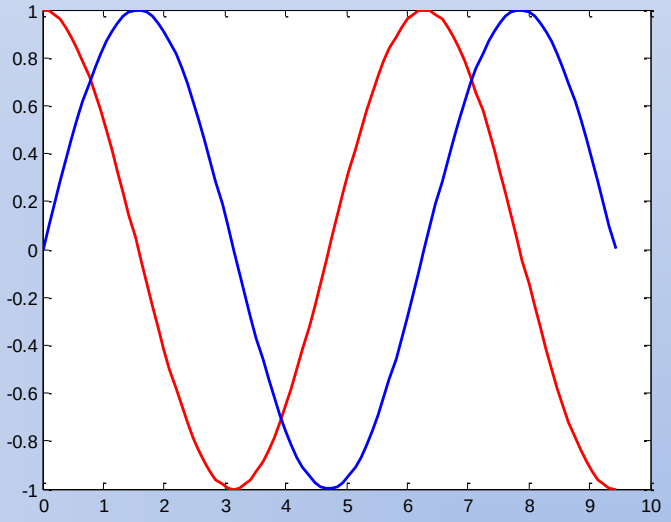
Multiple Graphs using 'hold'

By default, each plot command will erase previous plots.

The 'hold on' command will add plots to existing plots.

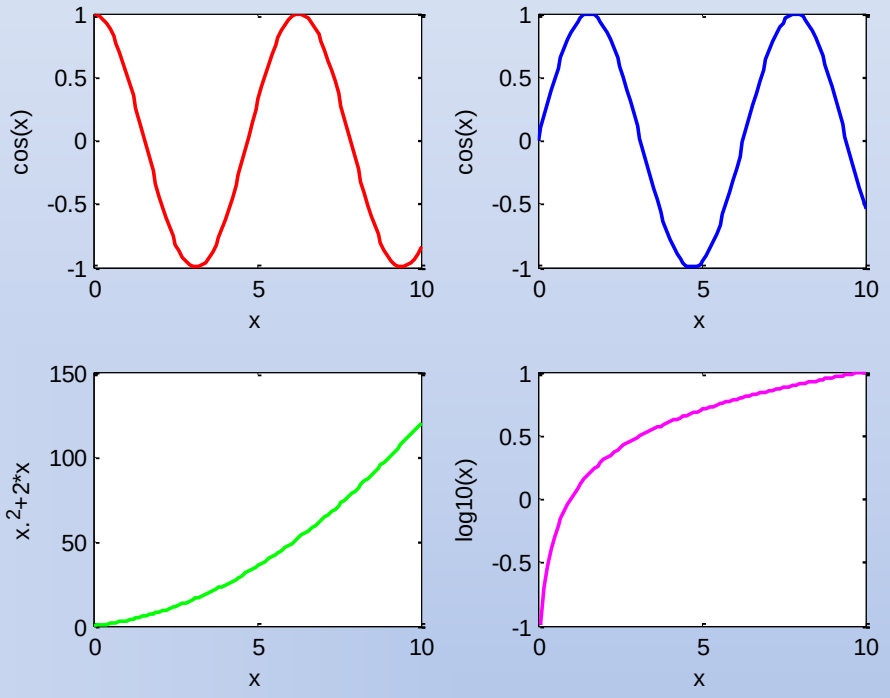
Example:

```
x=linspace(0,3*pi);
y1=cos(x);
y2=sin(x);
plot(x,y1,'r','linewidth',2)
hold on
plot(x,y2,'b','linewidth',2)
```



Multiple Graphs in Single Window - Subplot

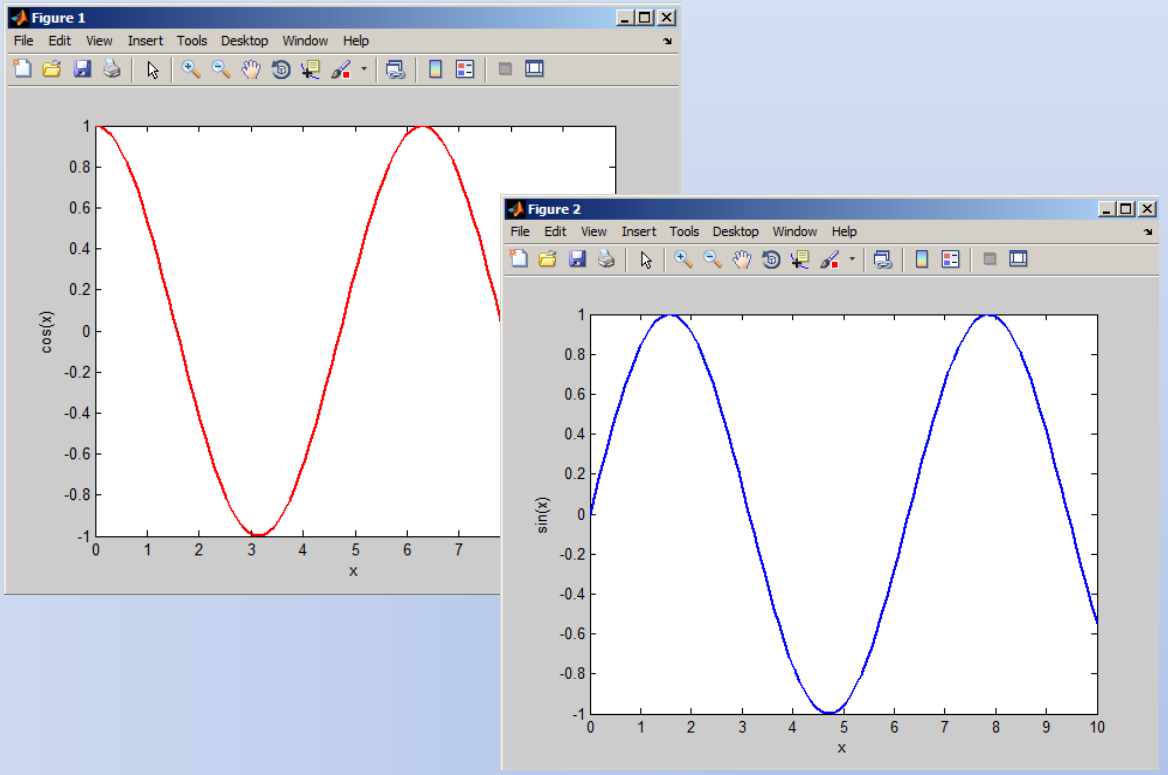
```
x=linspace(0,10);
y1=cos(x);
y2=sin(x);
y3=x.^2+2*x;
y4=log10(x);
%
subplot(2,2,1)
plot(x,y1,'r','linewidth',2)
xlabel('x')
ylabel('cos(x)')
%
subplot(2,2,2)
plot(x,y2,'b','linewidth',2)
xlabel('x')
ylabel('cos(x)')
%
subplot(2,2,3)
plot(x,y3,'g','linewidth',2)
xlabel('x')
ylabel('x.^2+2*x')
%
subplot(2,2,4)
plot(x,y4,'m','linewidth',2)
xlabel('x')
ylabel('log10(x)')
```



Argument list is:
rows, columns, subplot number

Multiple Figure Windows

```
x=linspace(0,10);
y1=cos(x);
y2=sin(x);
%
plot(x,y1,'r','linewidth',2)
%
figure
plot(x,y2,'b','linewidth',2)
%
figure(1)
xlabel('x')
ylabel('cos(x)')
%
figure(2)
xlabel('x')
ylabel('sin(x)')
```



Other Useful Commands

figure

- opens a new figure window, by default Figure number 1
- figure(n) creates new figure window (Figure n), or if Figure n exists, takes you to existing figure window n

text(x,y,'string')

- used to print text in the figure at location (x,y)

gtext('string')

- used to print text in the figure at location specified by mouse click

ginput(1)

- creates crosshairs on the screen
- returns (x,y) location of cursor at mouse click
- ginput(n) returns n pairs of locations

gtext and ginput example

```
clc; close all; clear
%
% A Simple Example using gtext
%
x=-3:0.3:3;
y=x.^3-5*x.^2+4;
plot(x,y)
xlabel('value of x')
ylabel('value of y')
title('A Simple Plotting Example')
% text(0,-30,'EGR106 example')
gtext('EGR106 example')
[x1,y1]=ginput(5);
x1
y1
```

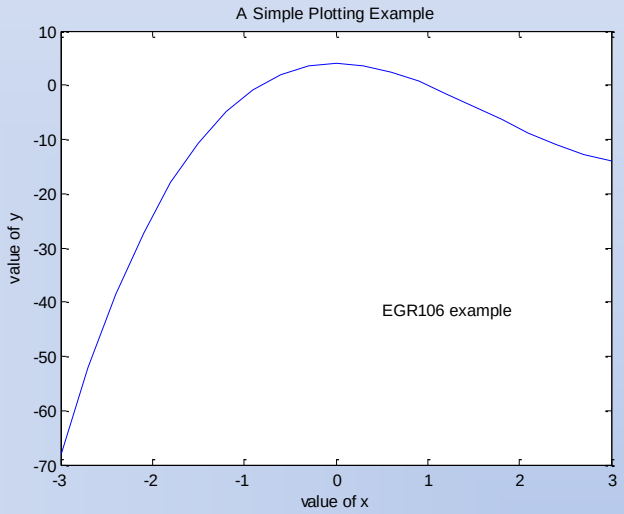
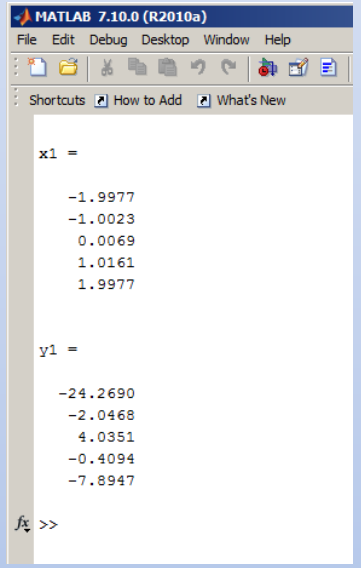


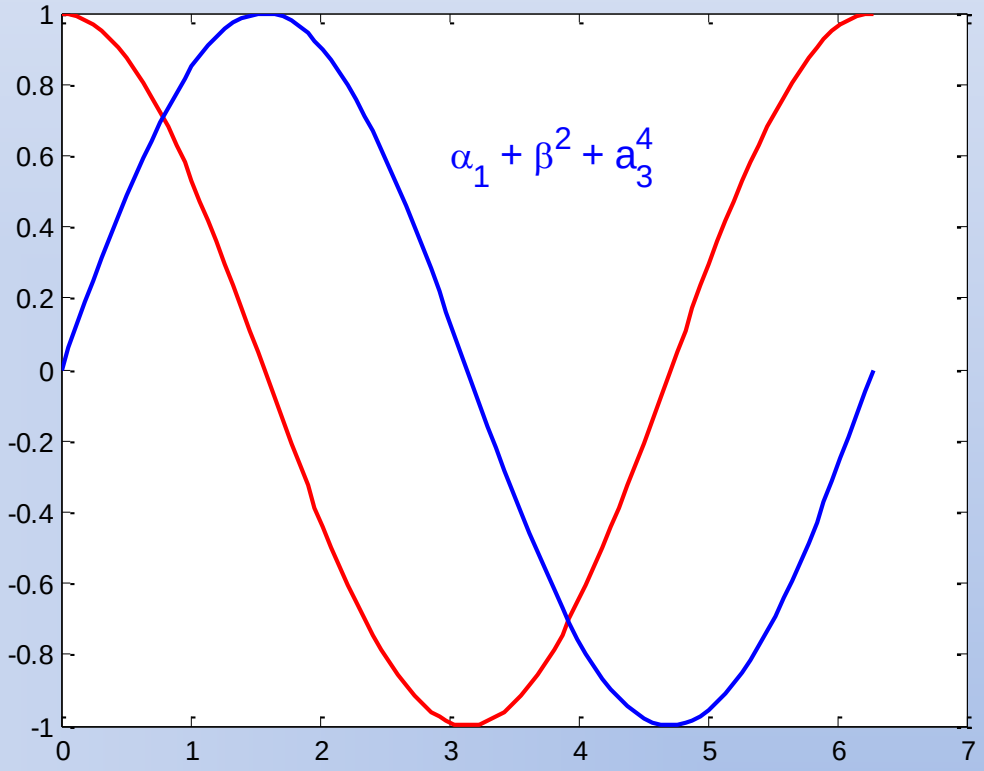
Figure Window



Command Window

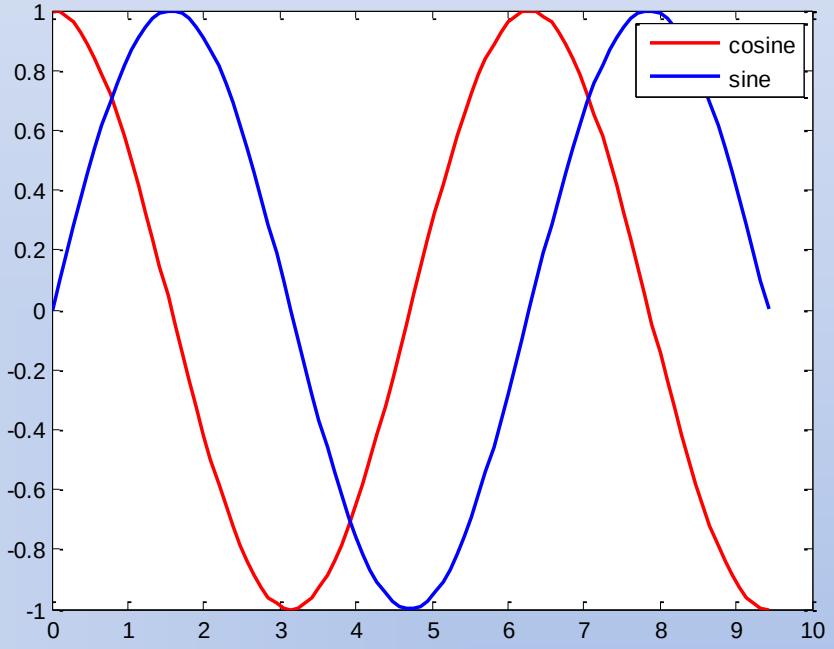
Adding Text with Greek Letters, Subscripts and Superscripts, Color

```
x=linspace(0,2*pi);
y1=cos(x);
y2=sin(x);
plot(x,y1,'r',x,y2,'b','linewidth',2)
text(3,.6,'\alpha_1 + \beta^2 + a_3^4', ...
    'fontsize',14,'color','b')
```



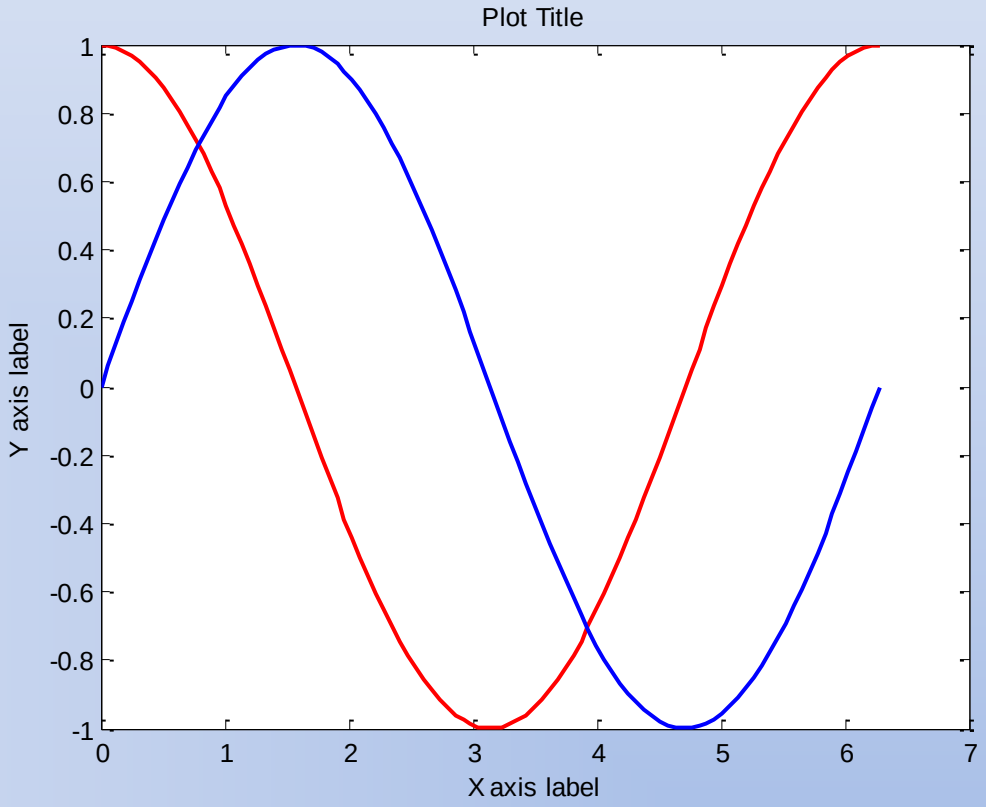
Adding a Legend

```
x=linspace(0,3*pi);
y1=cos(x);
y2=sin(x);
plot(x,y1,'r',x,y2,'b','linewidth',2)
legend('cosine','sine')
```



Adding Axis Labels

```
x=linspace(0,2*pi);
y1=cos(x);
y2=sin(x);
plot(x,y1,'r',x,y2,'b','linewidth',2)
xlabel('X axis label')
ylabel('Y axis label')
title('Plot Title')
```



Formatting Axes and Adding a Grid

Adding a grid:

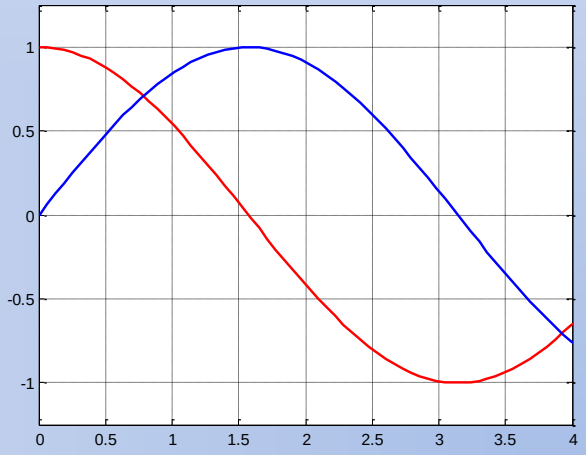
grid

Setting the axis limits:

axis([xmin xmax ymin ymax])

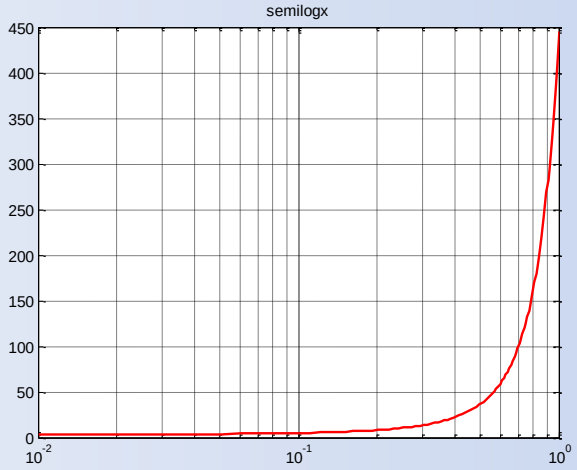
Example:

```
x=linspace(0,2*pi);
y1=cos(x);
y2=sin(x);
plot(x,y1,'r',x,y2,'b','linewidth',2)
axis([ 0 4 -1.25 1.25])
grid
```

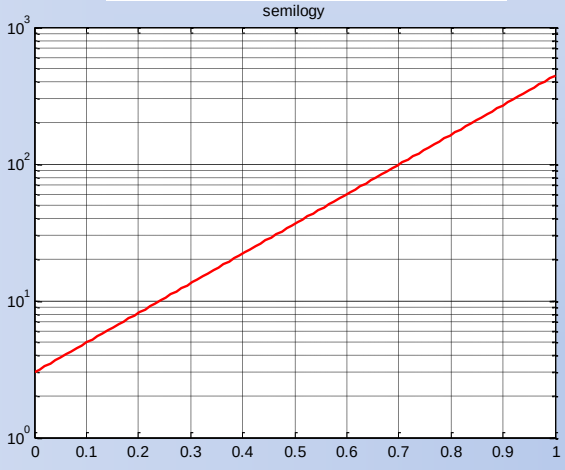


Log and Semilog Plots

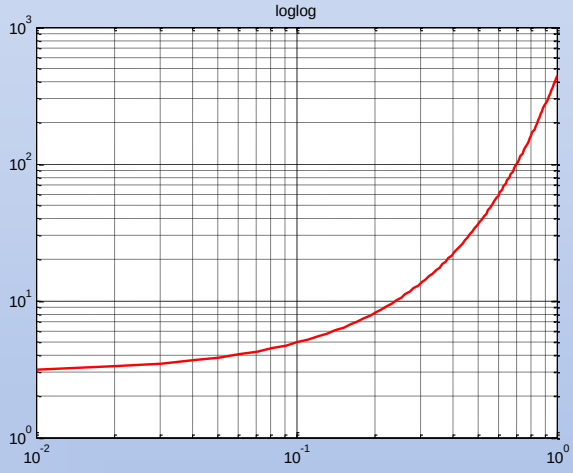
```
x=linspace(0,1);
y=3*exp(5*x);
semilogx(x,y,'r','linewidth',2)
grid
figure(2)
semilogy(x,y,'r','linewidth',2)
grid
figure(3)
loglog(x,y,'r','linewidth',2)
grid
title('Log and Semilog Plot')
```



semilogx(x,y)



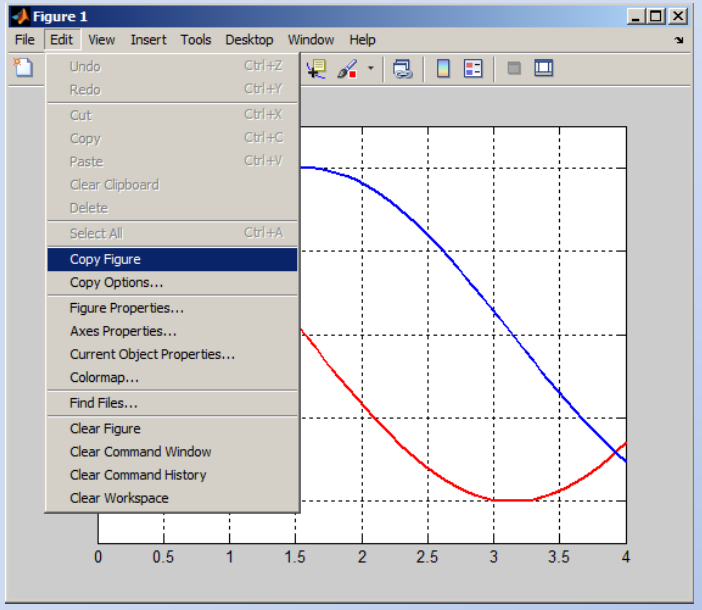
semilogy(x,y)



loglog(x,y)

Copying Figures to Other Applications

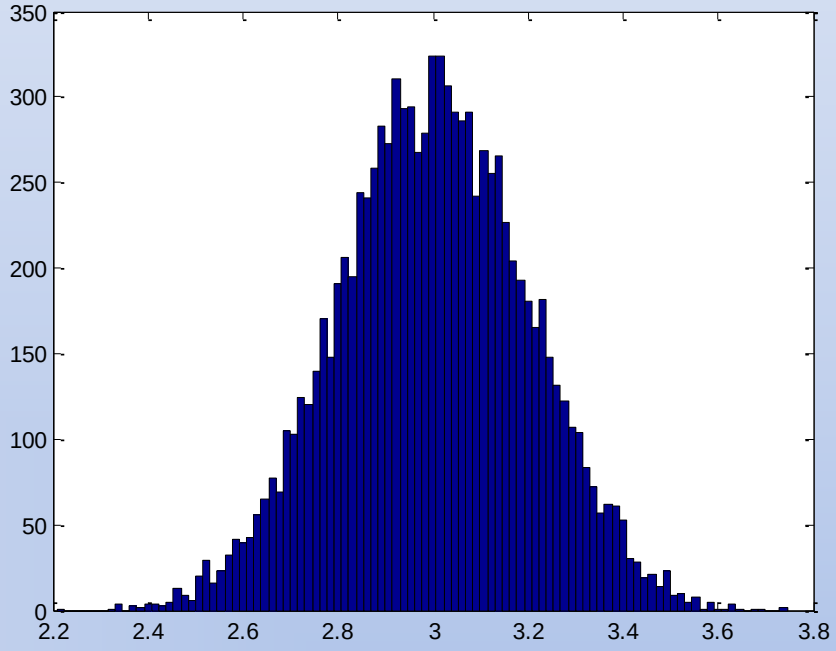
In Figure window => Edit => Copy



Other application (MS Word, Powerpoint, etc.) => Paste

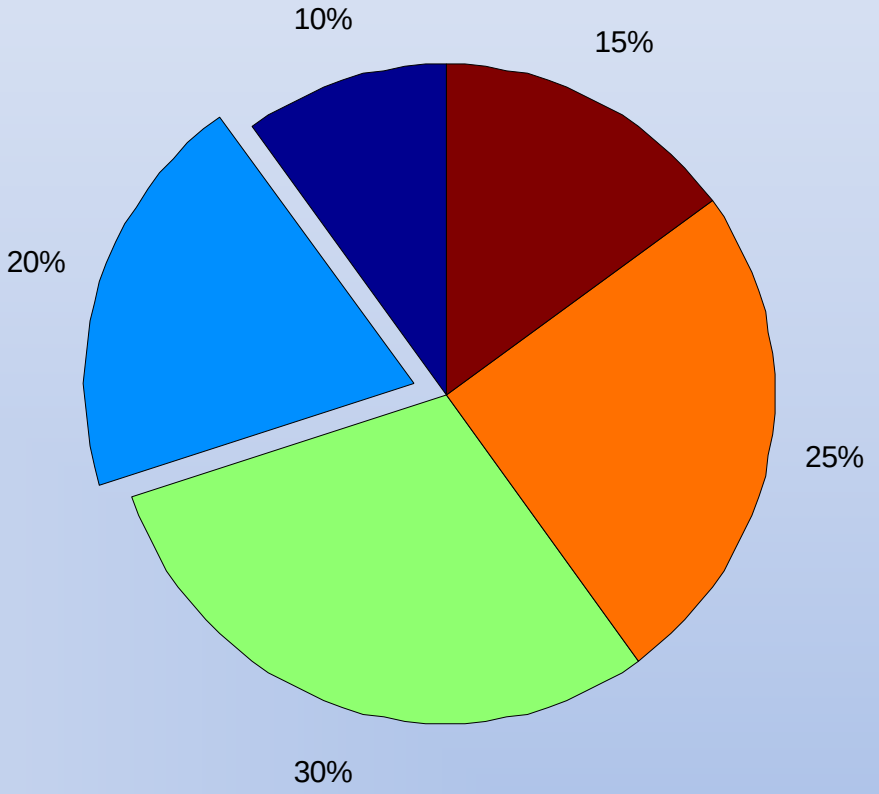
Histograms

```
% Normal (Gaussian) distribution of random numbers
%      with mean 3 and standard deviation 0.2
x=3+.2*randn(1,10000);
hist(x,100)
```



Pie Charts

```
x = [2 4 6 5 3];
explode = [0 1 0 0 0];
pie(x,explode)
```

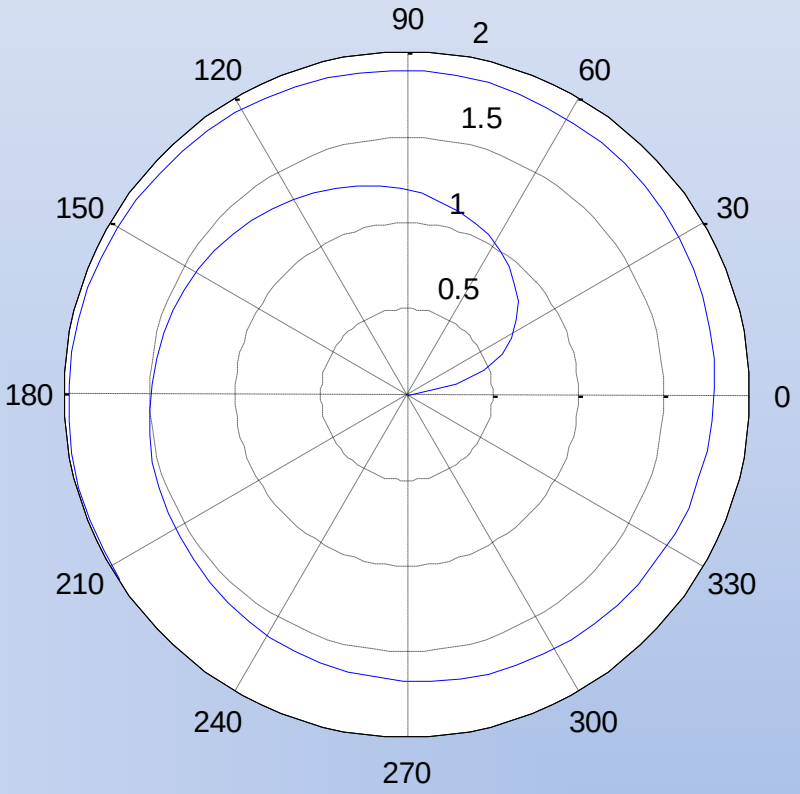


Polar Plots

```
x = 1:100;
r = log10(x);
t = x/10;
polar(t,r)
```

magnitude

angle in radians



Other Plot Types

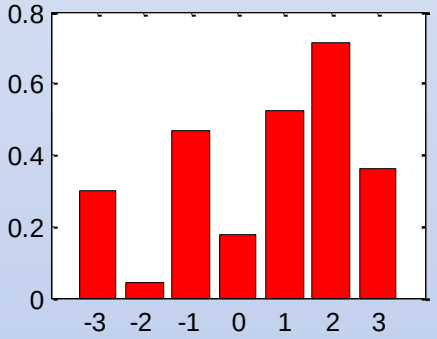
Vertical bar plot - bar

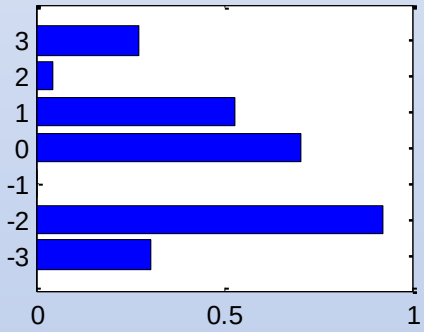
Horizontal bar plot - barh

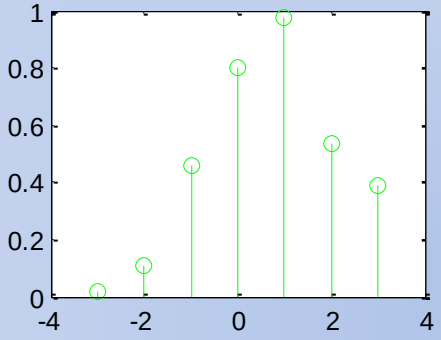
Stem plot - stem

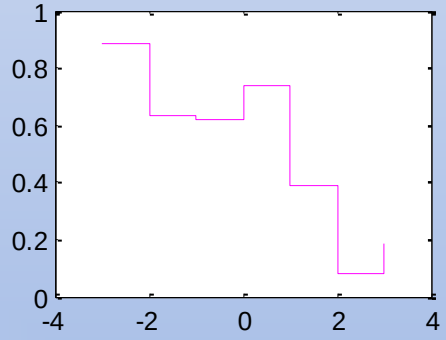
Stair plot – stairs

(see text for details)



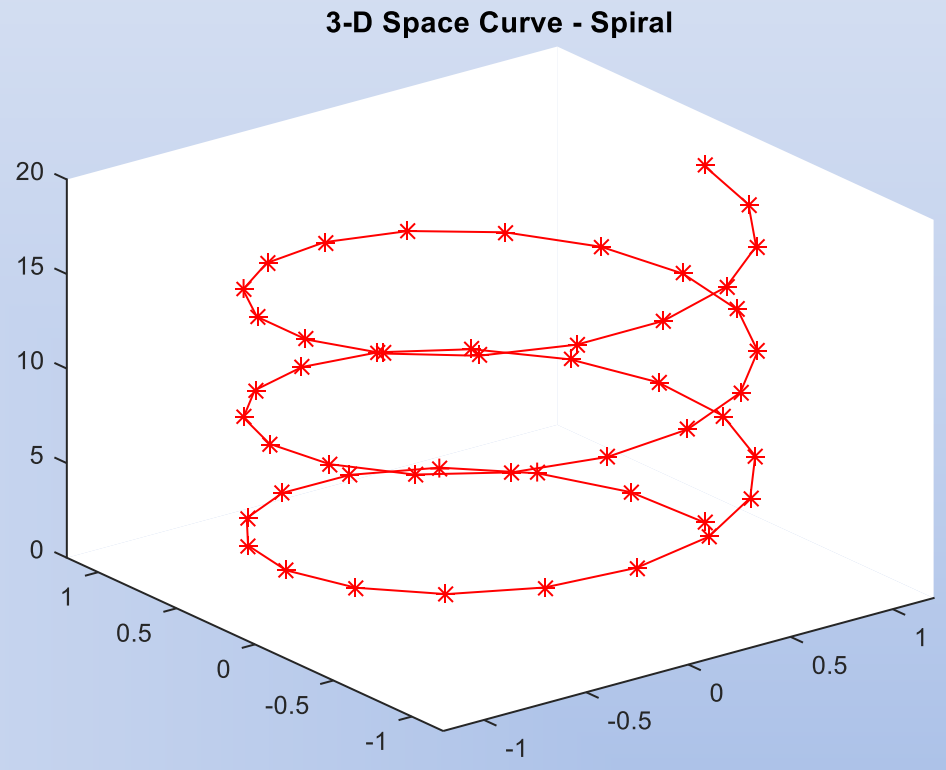






3D Space Curve Plot

```
t=linspace(0,6*pi,50);
x=cos(t);
y=sin(t);
z=t;
plot3(x,y,z,'r-*')
axis([-1.2 1.2 -1.2 1.2 0 20])
title('3-D Space Curve - Spiral')
pause; clear; close all
```



3D Surface and Contour Plots

```
x=-3:0.2:3;
y=x;
[X,Y]=meshgrid(x,y);
Z=1.8.^(-1.5*sqrt(X.^2+Y.^2)).*cos(0.5*Y).*sin(X);
surfc(X,Y,Z);
```