MATLAB: Additional Two-Dimensional and Three-Dimensional Plotting Functions

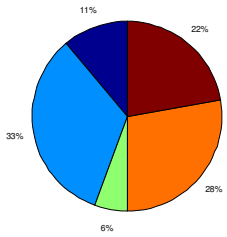
PhD Course Work 2014
Under the Faculty of Engineering

Instructor: Dr. G. L. Prajapati
IET-DAVV, Indore

More on 2-D Plot

Pie chart

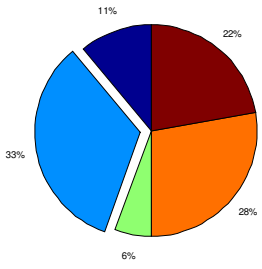
- x = [1 3 0.5 2.5 2];
- pie(x)



More on 2-D Plot

Pie chart

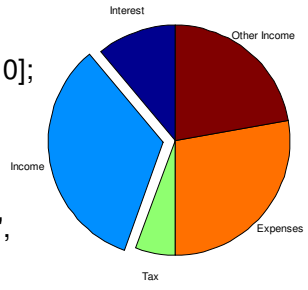
- x = [1 3 0.5 2.5 2];
- explode = [0 1 0 0 0];
- %offsets a slice from the pie
- pie(x,explode);



More on 2-D Plot

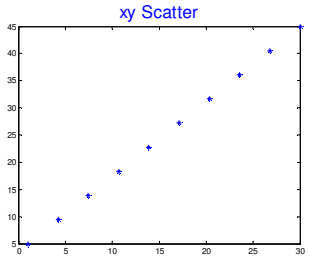
Pie chart

- x = [1 3 0.5 2.5 2];
- explode = [0 1 0 0 0];
- %offsets a slice from the pie
- pie(x,explode, {'Interest', 'Income', 'Tax', 'Expenses', 'Other Income'});



Plotting Data Points with No Line

If you specify a marker, but not a line style, only the markers are plotted.

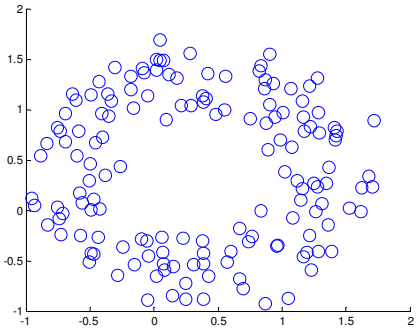


- x = linspace(1,30,10);
- y=linspace(5,45,10);
- plot(x,y,'b*');
- title('xy Scatter', 'FontSize', 20,'Color','Blue');

Scatter Plot

- It is used to plat a vector against the other with some markers. It requires at least two input vectors.
- theta = linspace(0,2*pi,150);
- x = sin(theta) + 0.75*rand(1,150);
- y = cos(theta) + 0.75*rand(1,150);
- a = 140;
- scatter(x,y,a,'o')

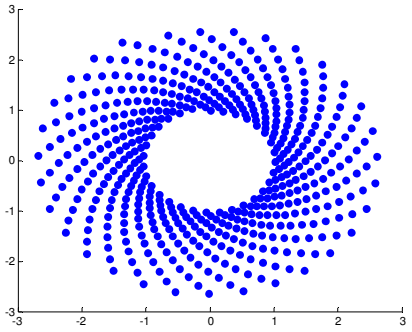
Output Figure



Create a scatter plot and return the scatter series object, s.

- theta = linspace(0,1,500);
- x = exp(theta).*sin(100*theta);
- y = exp(theta).*cos(100*theta);
- s = scatter(x,y,'Filled');
- s.LineWidth = 0.6;
- s.MarkerEdgeColor = 'b';

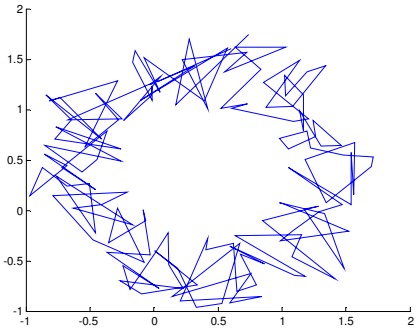
Output Figure



Line Plots

- line creates a line object in the current axes. You can specify the color, width, line style, and marker type, as well as other characteristics.
- theta = linspace(0,2*pi,150);
- x = sin(theta) + 0.75*rand(1,150);
- y = cos(theta) + 0.75*rand(1,150);
- line(x,y)

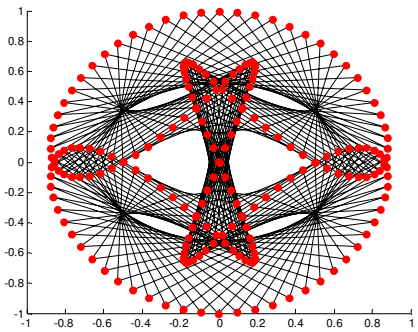
Output Figure (Line Plot)



line and scatter commands to draw complex figure

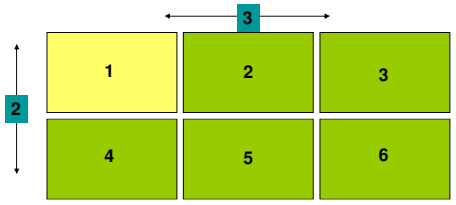
- t=-1:0.01:1;
- x=cos(57*pi*t).*sin(19*pi*t);
- y=sin(57*pi*t).*cos(38*pi*t);
- line(x,y,'Color', 'k');
- hold on
- scatter(x,y, 'r','filled')

Output Figure



Subplot Command

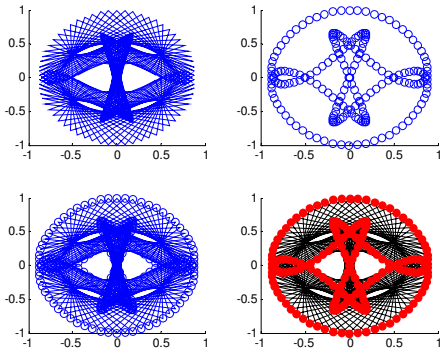
- There are times when it is better to create several smaller plots arranged in some kind of grid; `subplot(m,n,k)` does this…
  - m=rows, n=columns in the grid
  - k=current focus (numbered row-wise)
- Let's define a 2x3 subplot grid for: `subplot(2,3,1)` with the focus on the first plot.



Subplot Example

- t=-1:0.01:1;
- x=cos(57*pi*t).*sin(19*pi*t);
- y=sin(57*pi*t).*cos(38*pi*t);
- subplot(2,2,1);
- line(x,y);
- subplot(2,2,2);
- scatter(x,y);
- subplot(2,2,3);
- line(x,y);
- hold on
- scatter(x,y);
- subplot(2,2,4);
- line(x,y,'Color', 'k');
- hold on
- scatter(x,y, 'r','filled')

Subplot Example: Output Figures



Three-Dimensional Plots

MATLAB also includes a rich variety of three-dimensional plots that can be useful for displaying certain types data. It is generally useful for displaying two types of data:

- Two variables that are function of the same independent variable, when you wish to emphasize the importance of the independent variable.
- A single variable that is a function of two independent variables.

plot3

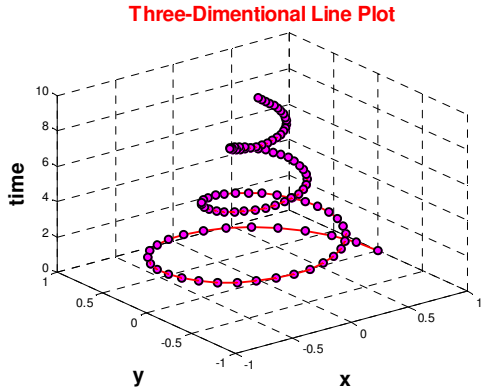
3-D line plot

- **Syntax**
- plot3(X1,Y1,Z1,...)
  plot3(X1,Y1,Z1,LineSpec,...)
  plot3(...,'*PropertyName*',PropertyValue,...)
  h = plot3(...)

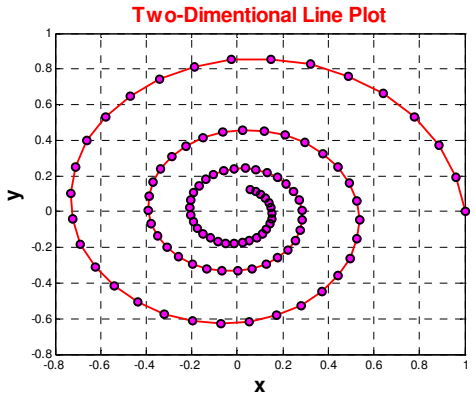
Example: plot3

```
t = 0:0.1:10;
x = exp(-0.2*t).*cos(2*t);
y = exp(-0.2*t).*sin(2*t);
plot3(x,y,t,'-o','LineWidth',1.5,'Color',
'Red','MarkerSize', 7,'MarkerEdgeColor',
'Black','MarkerFaceColor','Magenta');
title('\bfThree-Dimentional Line Plot', 'Color','Red',
'FontSize', 16);
xlabel('\bfx', 'Color','Black', 'FontSize', 16);
ylabel('\bfy','Color','Black', 'FontSize', 16);
zlabel('\bftime','Color','Black', 'FontSize', 16);
grid on;
```

Output Figure



```
plot(x,y,'-o','LineWidth',1.5,'Color', 'Red','MarkerSize',
7,'MarkerEdgeColor', 'Black','MarkerFaceColor','Magenta');
```



3-D Mesh Plot

- Connects a series of discrete data points with a mesh

mesh(x,y,z): This function creates a mesh or wireframe plot, where

- x is a two dimensional array containing the x values of every point to display,
- y is a two dimensional array containing the y values of every point to display, and
- z is a two dimensional array containing the z values of every point to display.

X and y arrays required for the plot can be created using the function meshgrid.

General for of meshgrid

- **[X , Y] = meshgrid(xstart:xinc:xend, ystart:yinc:yend)**
- [X,Y] = meshgrid(x,y)
  [X,Y] = meshgrid(x)
- **Examples**
- [X,Y] = meshgrid(1:3,10:14)

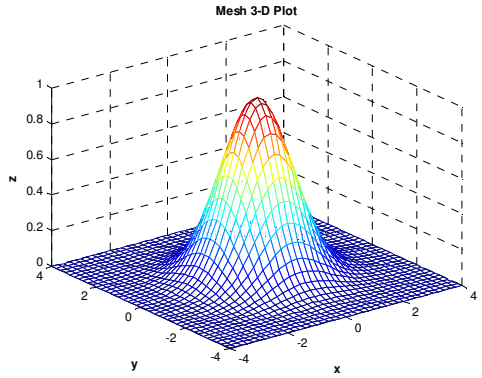
```
X =
     1     2     3
     1     2     3
     1     2     3
     1     2     3
     1     2     3
```

```
Y =
    10    10    10
    11    11    11
    12    12    12
    13    13    13
    14    14    14
```

Example [mesh plot]

- [x,y]=meshgrid(-4:0.2:4);
- z=exp(-0.5*(x.^2+y.^2));
- mesh(x,y,z);
- title('\bfMesh 3-D Plot');
- xlabel('\bfx');
- ylabel('\bfy');
- zlabel('\bfz');

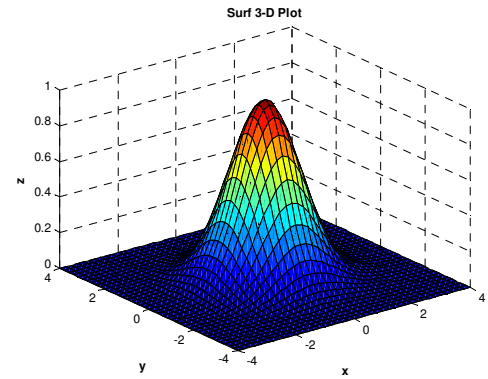
Output Figure



3-D Plots: Surf and Surfc

- Very similar to mesh, with filled grid
- surf and surfc create colored parametric surfaces specified by X, Y, and Z, with color specified by Z or C.
- [x,y]=meshgrid(-4:0.2:4);
- z=exp(-0.5*(x.^2+y.^2));
- surf(x,y,z);
- title('\bfSurf 3-D Plot');
- xlabel('\bfx');
- ylabel('\bfy');
- zlabel('\bfz');

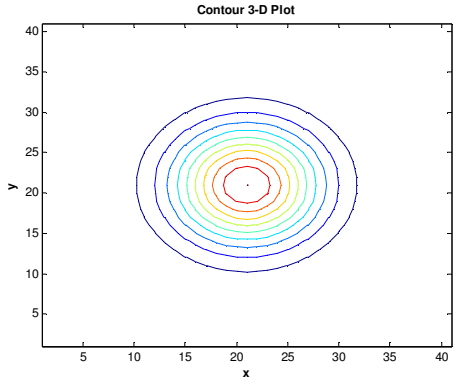
Output Figure



Contour

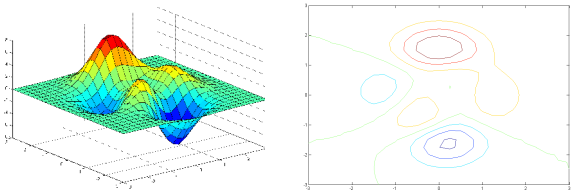
- [x,y]=meshgrid(-4:0.2:4);
- z=exp(-0.5*(x.^2+y.^2));
- contour(z);
- title('\bfContour 3-D Plot');
- xlabel('\bfx');
- ylabel('\bfy');
- zlabel('\bfz');

Output Figure



Contour

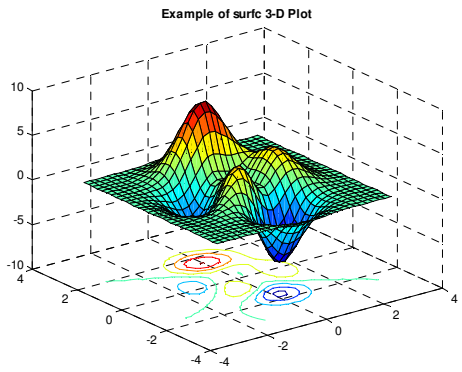
- Projection of equal heights of 3D plot onto a 2D plane
- Use like surf or mesh – contour(x,y,z)



Surfc Example

- [X,Y,Z] = peaks(30);
- surfc(X,Y,Z);
- title('\bfExample of surfc 3-D Plot');
- peaks is a function of two variables, obtained by translating and scaling Gaussian distributions, which is useful for demonstrating mesh, surf, contour, and so on.
- Z = peaks(n); returns an n-by-n matrix.
- [X,Y,Z] = peaks(...); returns two additional matrices, X and Y, for parametric plots

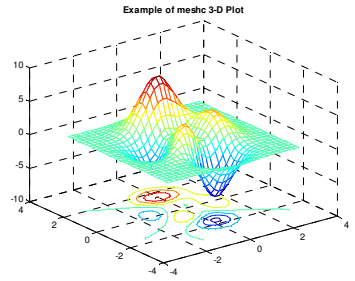
Output Figure



meshc

Combination mesh and contour plot

- [X,Y,Z] = peaks(30);
- meshc(X,Y,Z);
- title('\bfExample of meshc 3-D Plot');



Concluding Remarks

- Note that almost all 2D plotting commands such as xlabel, ylabel, title will still work in 3D plots
- 3D plots can be added as subplots just like any other plot type
- There are variety of 2D and 3D plots. Take the help of MATLAB to learn them for solving your problem.
- You can take variety of books or internet resource as well.