Table of Contents

CMS Tutorial

```
format long
pi % displays 3.141592653589793
format short
pi % displays 3.1416

%clc

format compact

ans =
   3.141592653589793
ans =
    3.1416
```

Matrices ---- http://www.math.harvard.edu/computing/matlab/index.html

```
a = [1 2 3; 4 5 6; 2 3 7];
a
a'
det(a)
inv(a)
poly(a)
eig(a)

ones(4,7)
I = eye(3)
I(:,2)

a =
     1     2     3
     4     5     6
     2     3     7
ans =
     1     4     2
```

```
     2     5     3
     3     6     7
ans =
    -9
ans =
   -1.8889    0.5556    0.3333
    1.7778   -0.1111   -0.6667
   -0.2222   -0.1111    0.3333
ans =
    1.0000  -13.0000   15.0000    9.0000
ans =
   11.6456
   -0.4325
    1.7869
ans =
     1     1     1     1     1     1     1
     1     1     1     1     1     1     1
     1     1     1     1     1     1     1
     1     1     1     1     1     1     1
I =
     1     0     0
     0     1     0
     0     0     1
ans =
     0
     1
     0
```

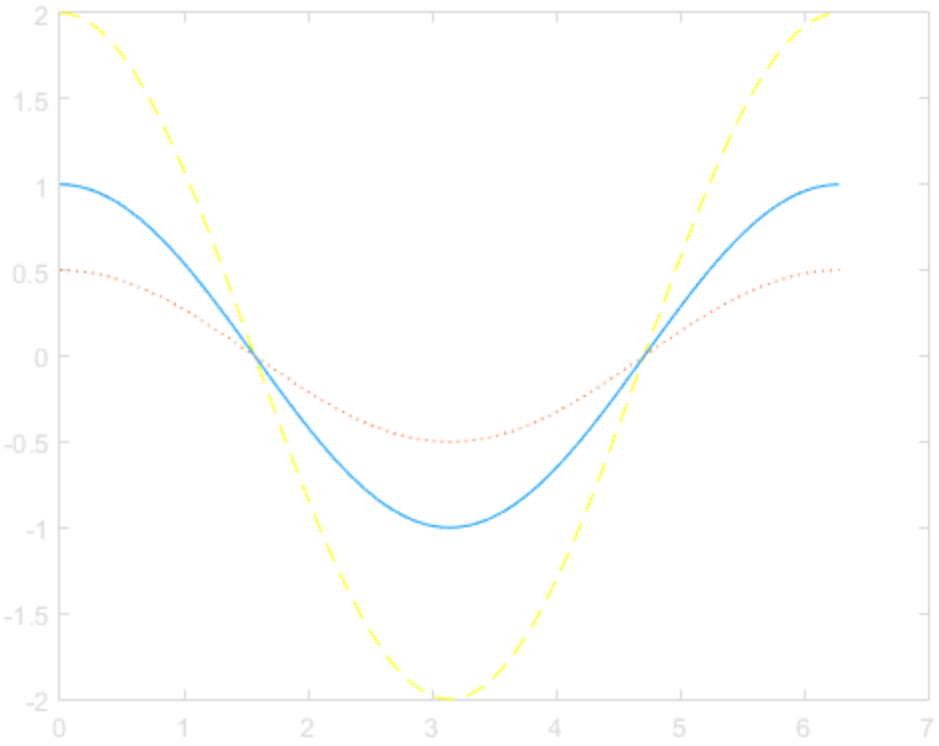
Dealing with Functions

```
syms x
int(1/ (1 + x^2) )

quad('sin(x)-x',0,pi)

diff('tan(x)*x')

ans =
atan(x)
ans =
   -2.9348
ans =
   -19    13   -70    80   -79     1    78
```

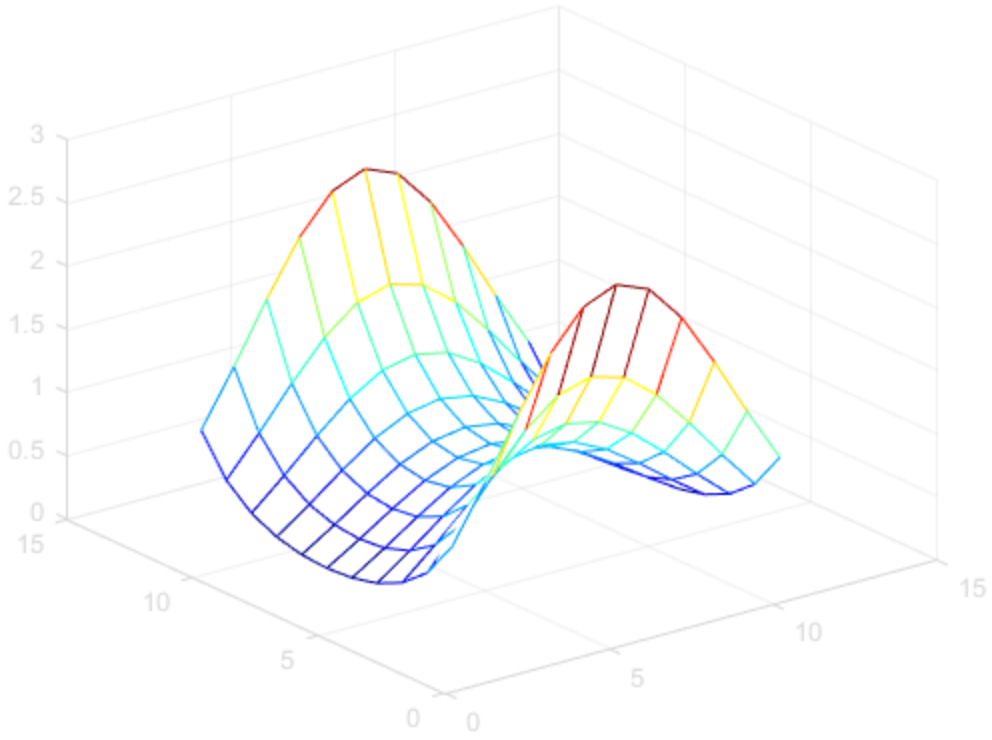
plotting functions

```
fplot('sin',[0,10])

t=0:0.1:10; y=sin(t); plot(t,y)


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])
plot(x,y1,'--',x,y2,'-',x,y3,':')
```




```
[x,y]=meshgrid(-1:.2:1,-1:.2:1);
z=exp(-x.^2+y.^2);
mesh(z)
```

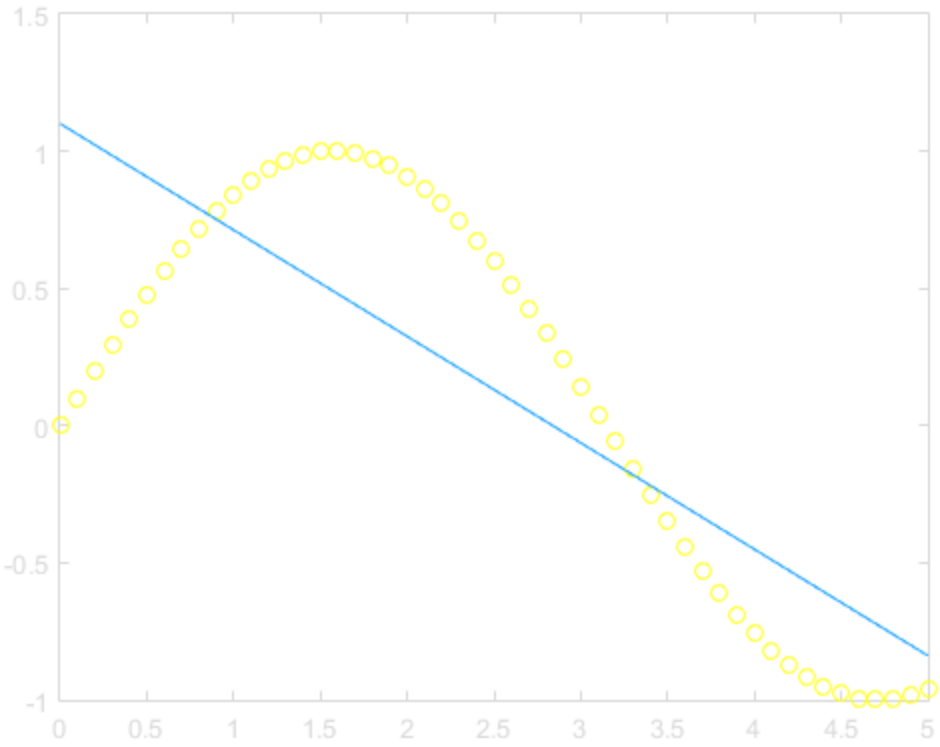


visualize data

```
a=rand(100); b=inv(a); imagesc(b);
a=magic(100); imagesc(a);

roots([9 2 3 4 5 7])

fzero('cos(x)-x',0.5)

[B,V]=bucky; gplot(B,V);
```

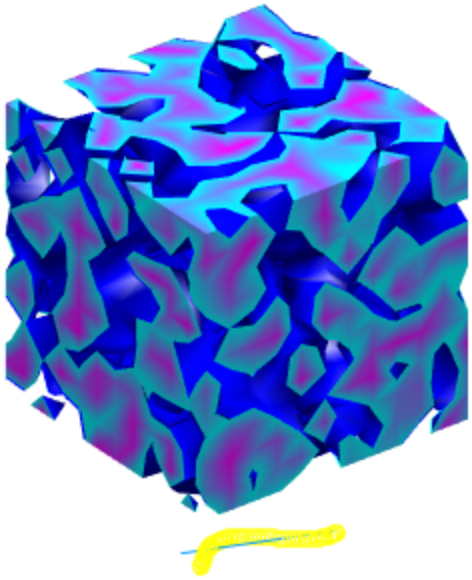
```
ans =
   0.6860 + 0.7747i
   0.6860 - 0.7747i
  -0.8839 + 0.0000i
  -0.3551 + 0.8341i
  -0.3551 - 0.8341i
ans =
    0.7391
```

```
x=(0:0.1:5)';
y=sin(x);
p=polyfit(x,y,1);
f=polyval(p,x);
plot(x,y,'o',x,f,'-');
```



some more advanced graphics

```
data = rand(12,12,12);
isoval = .4;
h = patch(isosurface(data,isoval),...
    'FaceColor','blue',...
    'EdgeColor','none',...
    'AmbientStrength',.2,...
    'SpecularStrength',.7,...
    'DiffuseStrength',.4);
isonormals(data,h)
patch(isocaps(data,isoval),...
    'FaceColor','interp',...
    'EdgeColor','none')
colordef black;
colormap cool;
daspect([1,1,1]);
axis off; view(3);
camlight right;
camlight left;
set(gcf,'Renderer','zbuffer');
material shiny;
lighting phong;
```

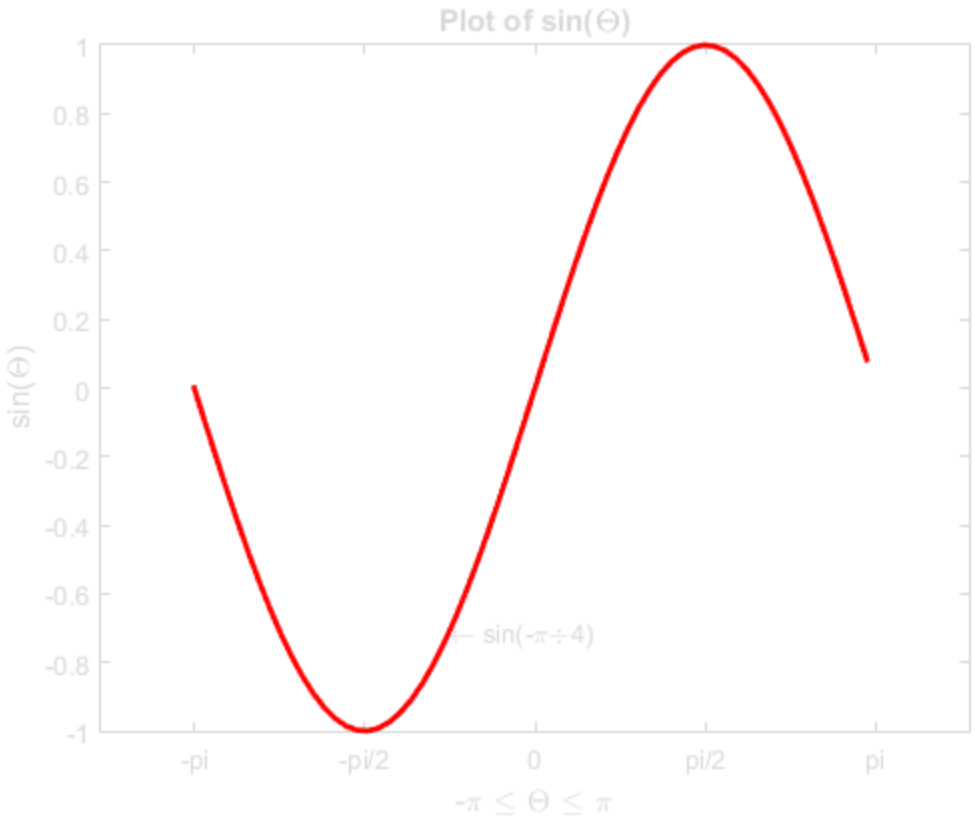


More plotting ---- https://wiki.harvard.edu/confluence/display/USERDOCS/Matlab+Tutorial

```
x = -pi:.1:pi;
y = sin(x);
p = plot(x,y)

p =
  Line with properties:

              Color: [1 1 0.0667]
          LineStyle: '-'
          LineWidth: 0.5000
             Marker: 'none'
         MarkerSize: 6
    MarkerFaceColor: 'none'
              XData: [1x63 double]
              YData: [1x63 double]
              ZData: [1x0 double]

  Use GET to show all properties
```

gca returns the handle to the current axes for the current figure.

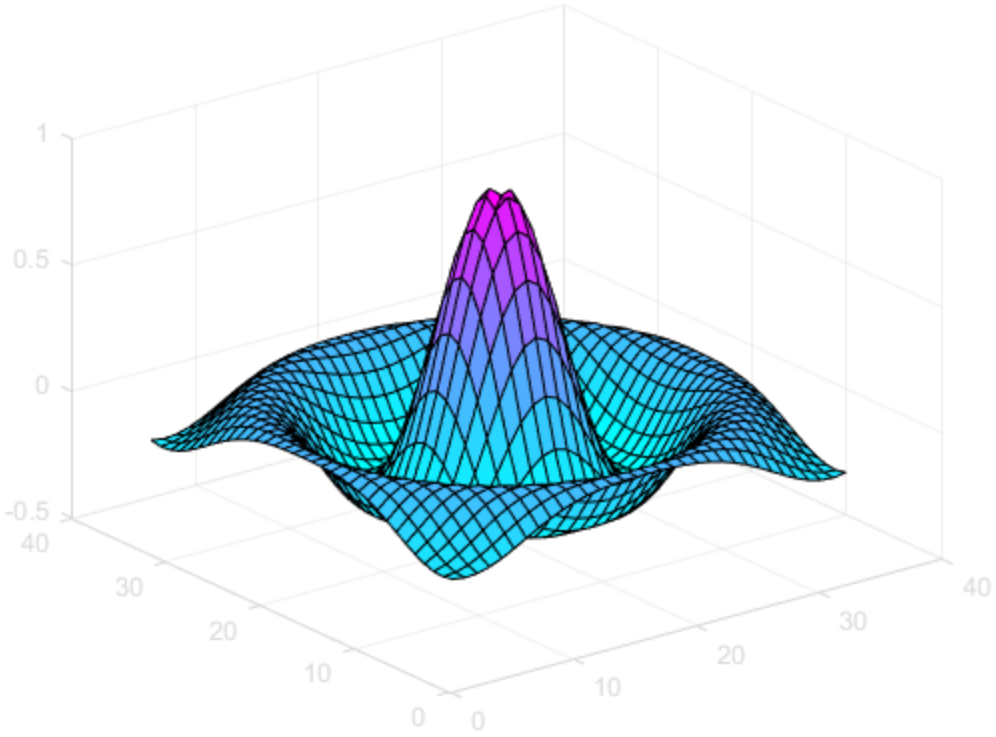
```
set(gca,'XTick',-pi:pi/2:pi)
set(gca,'XTickLabel',{'-pi','-pi/2','0','pi/2','pi'})

% \pi, \leq, \Theta are all from "latex" typesetting system.

xlabel('-\pi \leq \Theta \leq \pi')
ylabel('sin(\Theta)')
title('Plot of sin(\Theta)')

% \Theta appears as a Greek symbol (see String)
% Annotate the point (-pi/4, sin(-pi/4))

text(-pi/4,sin(-pi/4),'\leftarrow sin(-\pi\div4)',... % Notice the
 line continuation with ellipsis (...)
     'HorizontalAlignment','left')

% Change the line color to red and
% set the line width to 2 points

set(p,'Color','red','LineWidth',2)
```



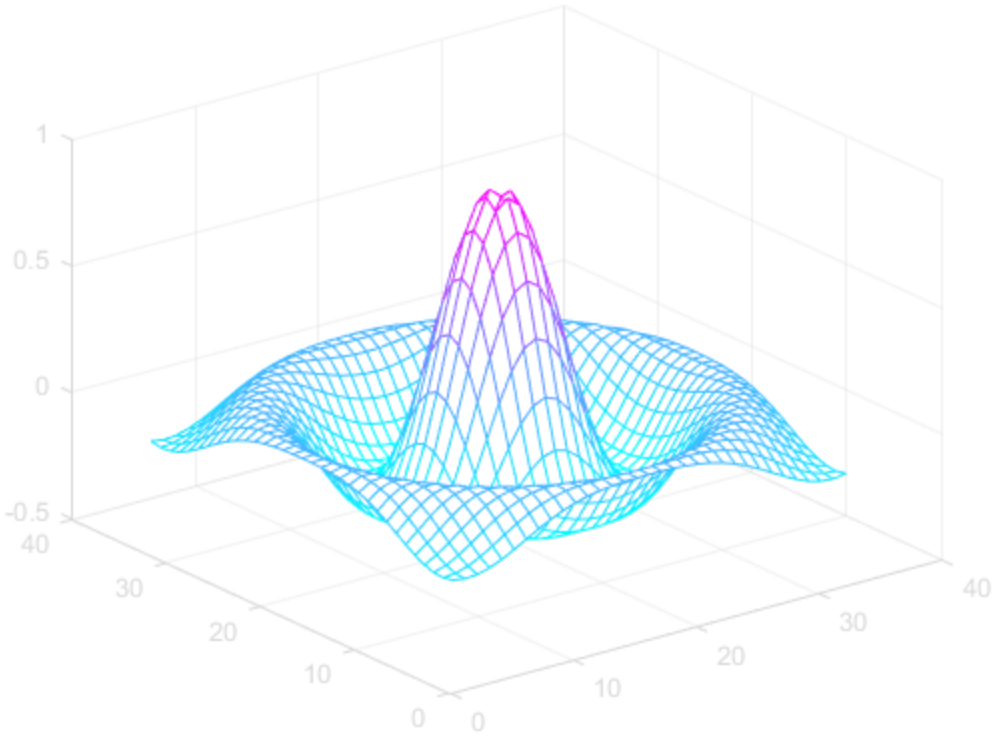

```
%If we have two arrays that define two orthogonal axes, matlab makes
 it easy to
%produce a grid from the axes to form the basis for plotting.
%This is done with the meshgrid command. Let's define the two axes and
 see how the meshgrid command works.

x=[-8:4:8]; % an array of points on the x-axis [-8 -4 0 4 8]
y=[-8:4:8]; % an array of points on the y-axis [-8 -4 0 4 8]
[X Y]=meshgrid(x,y) % returns two matrices X and Y:


% Note (X[j],Y[j]) define points in the 2D space defined by the x and
 y axes.
% Thus, if we have a function defined on x and y, i.e. f(x,y),
% evaluation of the function over all the ordered pairs of points
% in x and y can now use the whole matrices X and Y instead of looping
% over all the points. See below for an example of how this is done in
 the
% context of a 3D plot below.

% Plot of 3D sinc function.
x=[-8:.5:8];
y=[-8:.5:8];
% Look up "meshgrid" in matlab document
% Returns rectangular grid
[X Y]=meshgrid(x,y);
```

```
R = sqrt(X.^2 + Y.^2); % Notice how element-wise operation is used.
Z = sin(R)./R;  % Sinc function
surf(Z); % surface plot
```

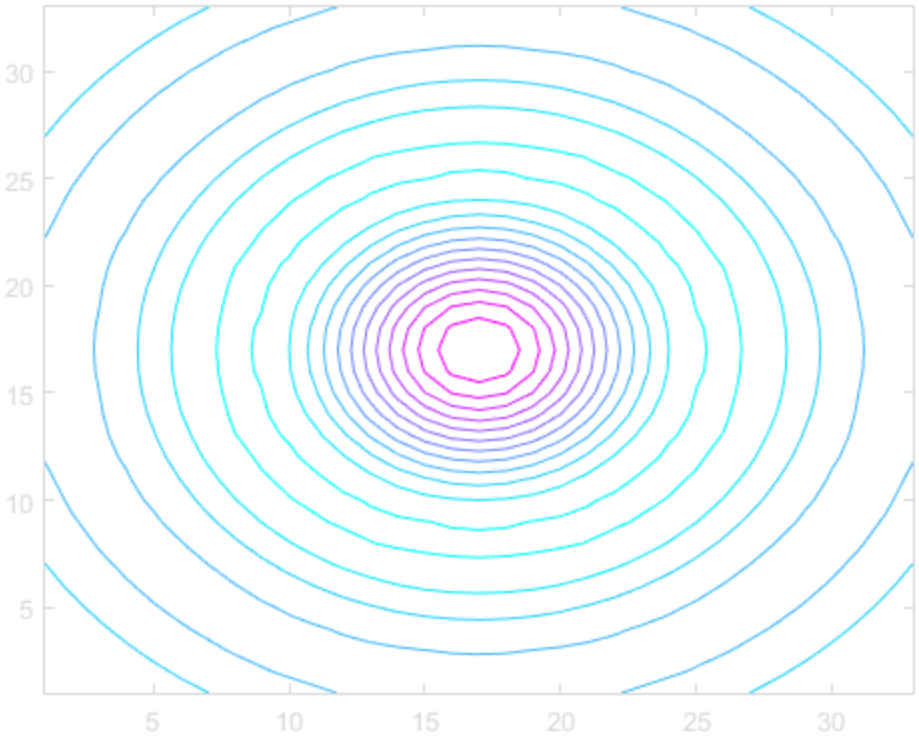
```
X =
    -8    -4     0     4     8
    -8    -4     0     4     8
    -8    -4     0     4     8
    -8    -4     0     4     8
    -8    -4     0     4     8
Y =
    -8    -8    -8    -8    -8
    -4    -4    -4    -4    -4
     0     0     0     0     0
     4     4     4     4     4
     8     8     8     8     8
```




```
mesh(Z); % mesh plot
```




```
contour(Z); % contour plot
```



Referenced Sites

```
%http://www.math.harvard.edu/computing/matlab/index.html
%https://www.mccormick.northwestern.edu/documents/students/
undergraduate/introduction-to-matlab.pdf
%https://wiki.harvard.edu/confluence/display/USERDOCS/Matlab+Tutorial

% other good stuff
%https://web.eecs.umich.edu/~aey/eecs451/matlab.pdf
%http://www.nmr.mgh.harvard.edu/~ona/matlab.html
%https://wiki.harvard.edu/confluence/pages/viewpage.action?
pageId=162432143
```

Published with MATLAB® R2015a