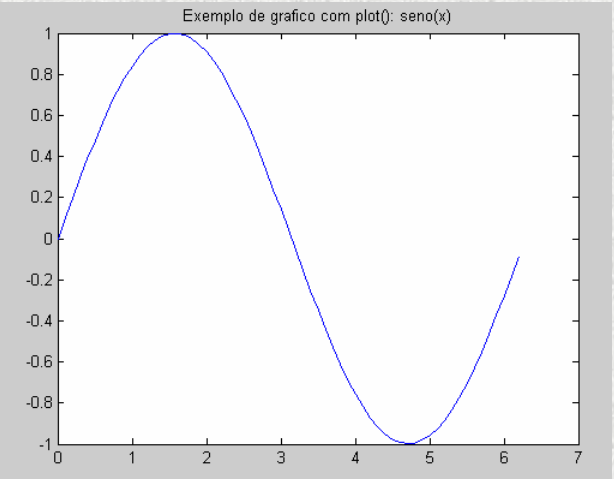
Notas de Aula
MatLab - Gráficos

Routo Terada
www.ime.usp.br/~rt
Depto. C. da Computação - USP

Bibliografia:
D. Hanselman et al., MatLab 5 -- Guia do Usuário,
Editora Makron 1999

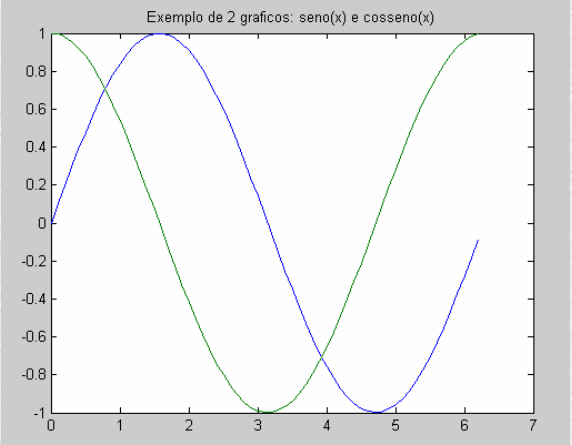
plot()

```
x=0:0.1:2*pi; % define pontos no eixo x
y=sin(x); % seno de x
plot(x,y)
title('Exemplo de grafico com plot(): seno(x)') % define título
```



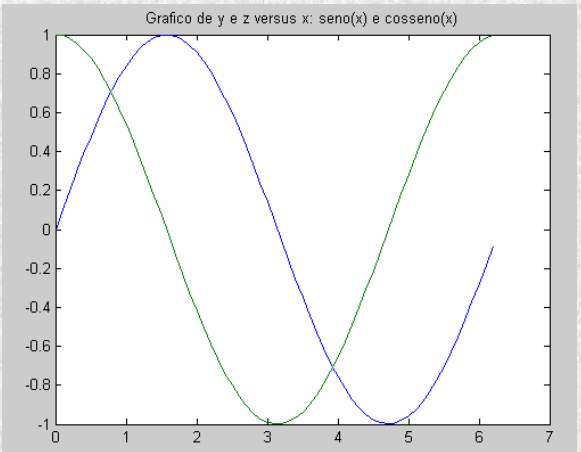
plot()

```
x=0:0.1:2*pi; % define pontos no eixo x
y=sin(x); % seno de x
z=cos(x) % cosseno de x
plot(x,y,x,z) % dois gráficos
title('Exemplo de 2 graficos: seno(x) e cosseno(x)') % define título
```



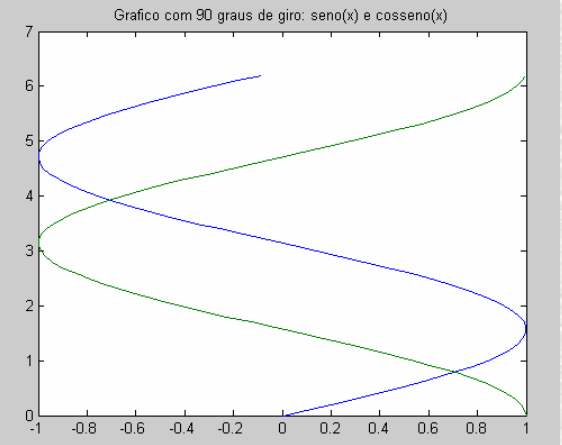
plot()

```
x=0:0.1:2*pi; % define pontos no eixo x
y=sin(x); % seno de x
z=cos(x); % cosseno de x
Matr=[y;z]; % definir uma matriz com seno e cosseno
plot(x,Matr) %  // gráfico de Matr versus x
title('Grafico de y e z de Matr versus x: seno(x) e cosseno(x)')
                                        % define título
```



plot()

```
x=0:0.1:2*pi; % define pontos no eixo x
y=sin(x); % seno de x
z=cos(x); % cosseno de x
Matr=[y;z] % definir uma matriz com seno e cosseno
plot(Matr,x) %  // matriz como 1o. argumento
title('Grafico com 90 graus de giro: seno(x) e cosseno(x)')
                                        % define título
```



plot()

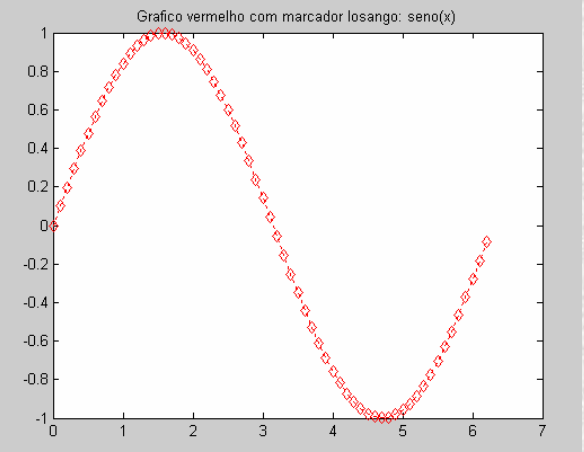
símbolo	cor
b	azul
g	verde
r	vermelho
c	ciano
m	magenta
y	amarelo
k	preto
w	branco

símbolo	marca
.	ponto
o	círculo
x	xis
s	quadrado
d	losango
v	triâng p/ baixo
^	triâng p/ cima
p	pentagrama
h	hexagrama
<	Triâng p/ esq.
>	Triâng p/ dir.

símbolo	Tipo de linha
-	contínua
:	pontilhada
-.	Traço e pto.
--	tracejada

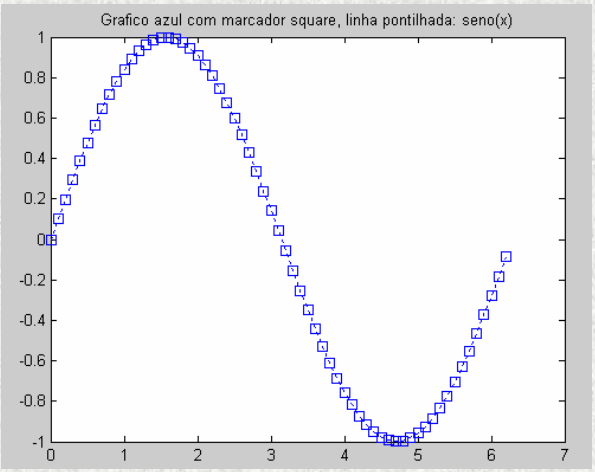
plot()

```
x=0:0.1:2*pi; % define pontos no eixo x
y=sin(x); % seno de x
z=cos(x); % cosseno de x
plot(x,y,'rd') % r de red, e d de losango
title('Grafico vermelho com marcador losango: seno(x)')
                                        % define título
```



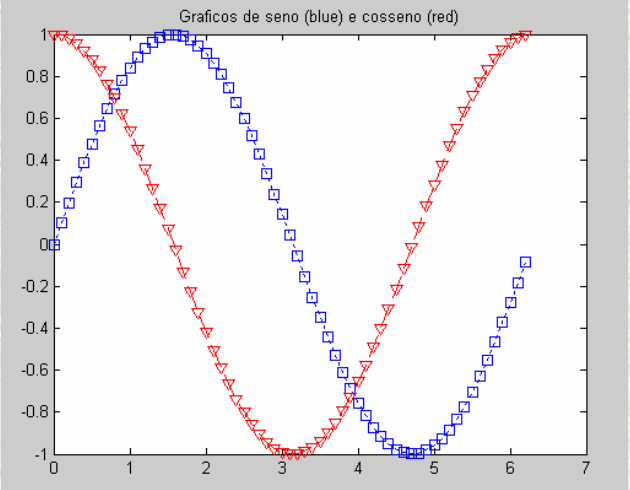
plot()

```
x=0:0.1:2*pi; % define pontos no eixo x
y=sin(x); % seno de x
z=cos(x); % cosseno de x
plot(x,y,'b:s')
title('Grafico azul com marcador square, linha pontilhada: seno(x)')
```



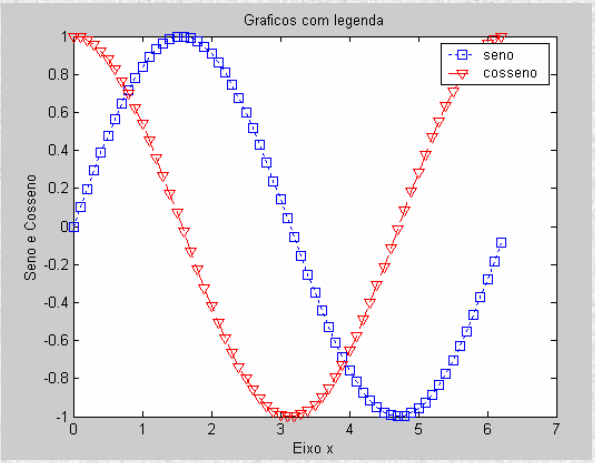
plot()

```
x=0:0.1:2*pi; % define pontos no eixo x
y=sin(x); % seno de x
z=cos(x); % cosseno de x
plot(x,y,'b:s',x,z,'rv-')
title('Graficos de seno (blue) e cosseno (red)')
```



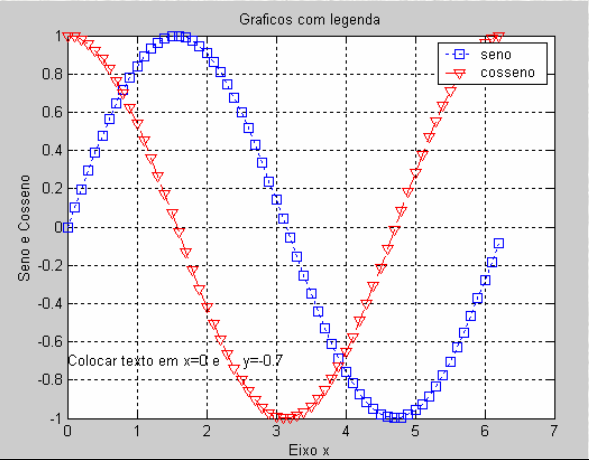
plot()

```
x=0:0.1:2*pi; % define pontos no eixo x
y=sin(x); z=cos(x);
plot(x,y,'b:s',x,z,'rv--')
title('Graficos com legenda')
xlabel('Eixo x') % eixo horizontal
ylabel('Seno e Cosseno') % eixo vertical
legend('seno','cosseno') % inserir legenda, na ordem
%       que pode ser deslocada arrastando-a c/ mouse
```



plot()

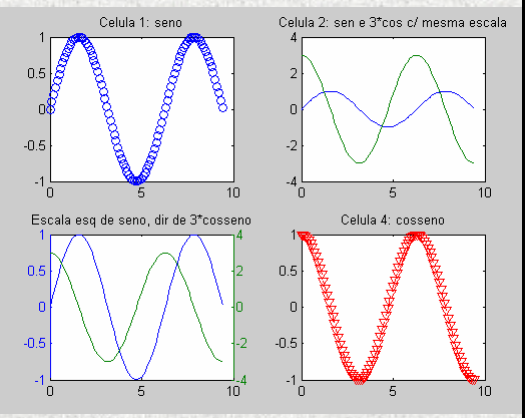
```
x=0:0.1:2*pi; % define pontos no eixo x
y=sin(x); z=cos(x);
plot(x,y,'b:s',x,z,'rv--')
title('Graficos com legenda')
xlabel('Eixo x') % eixo horizontal
ylabel('Seno e Cosseno') % eixo vertical
legend('seno','cosseno') % inserir legenda, na ordem
grid on  % para mostrar reticulado; grid off p/ apagar
text(0,-0.7,'Colocar texto em x=0 e     y=0.7') % p/ incluir texto
```



subplot()

subplot(m,n,prox) divide a janela de gráficos em m linhas e n colunas, sendo prox a próxima célula a receber o gráfico

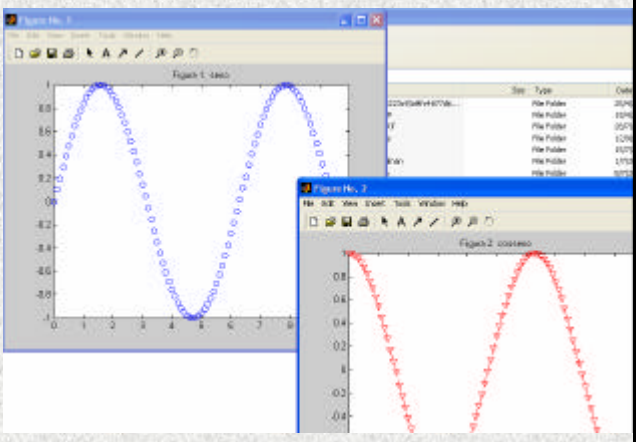
```
x=0:0.1:3*pi; % define pontos no eixo x
y=sin(x); % seno de x
z=cos(x); % cosseno de x
w=3*cos(x);
%%%%%%%%%%
subplot(2,2,1)
plot(x,y,'bo')
title('Celula 1: seno')
%%%%%%%%%%
subplot(2,2,4)
plot(x,z,'rv-')
title('Celula 4: cosseno')
%%%%%%%%%%
subplot(2,2,2)
plot(x,y,x,w)
title('Celula 2: sen e 3*cos c/ mesma escala')
%%%%%%%%%
subplot(2,2,3)
% plotyy p/ ter escala distinta nos eixos verticais
% escala de y no eixo vert esquerdo,
% de w no vert direito
plotyy(x,y,x,w)
title('Escala esq de seno, dir de 3*cosseno')
```



plotyy()

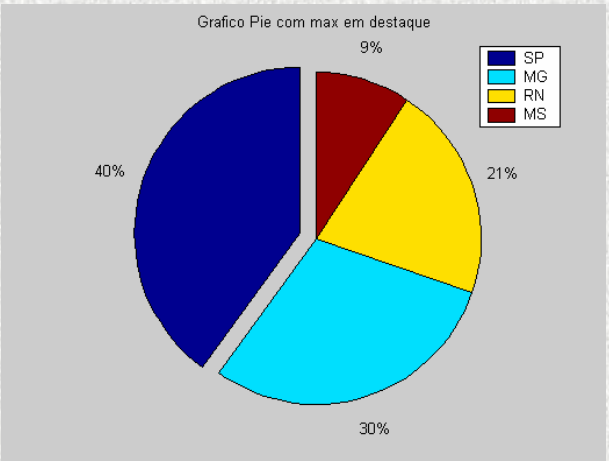
```
y=sin(x); % seno de x
z=cos(x); % cosseno de x
%%%%%%%%%%%%%%%%%%%%%%
figure(1) % próx gráfico na janela 1
plot(x,y,'bo')
title('Figura 1: seno')
%%%%%%%%%%%%%%%%%%%%%%
figure(2) % próx gráfico na janela 2
plot(x,z,'rv--')
title('Figura 2: cosseno')
```

figure()



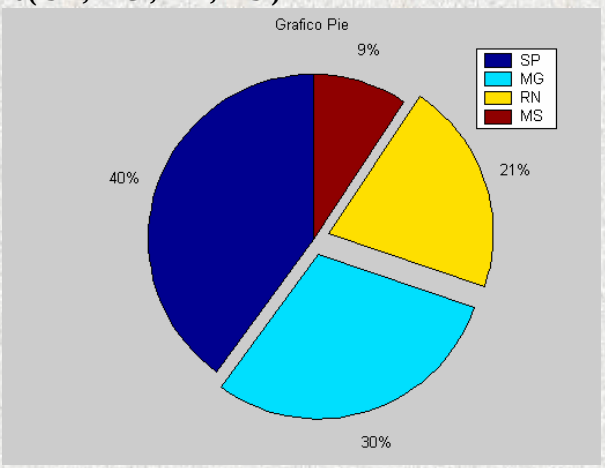
pie()

```
A=[4.3 3.2 2.25 1];
pie(A,A==max(A)); % destaca a fatia maior
title('Grafico Pie com max em destaque')
legend('SP','MG','RN','MS')
```



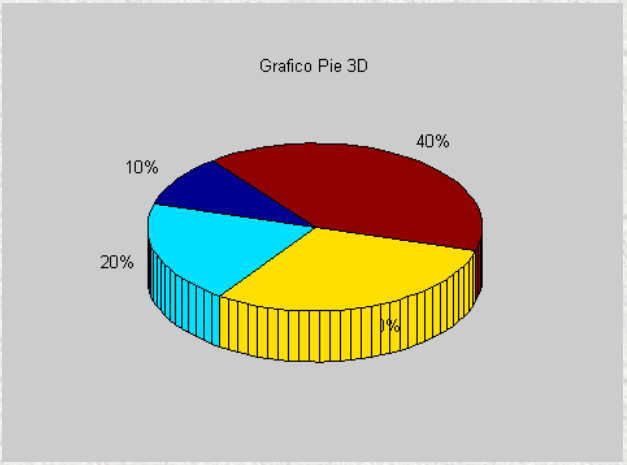
pie()

```
A=    [4.3 3.2 2.25 1];
pie(A,[0    1   1     0]) % destaca as fatias com 1 na posicao correspondente
title('Grafico Pie')
legend('SP','MG','RN','MS')
```



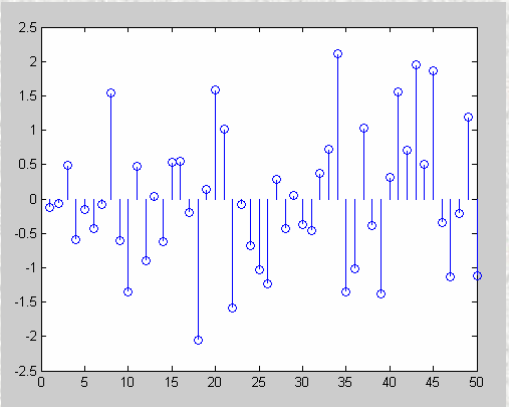
pie3()
p/ 3D

```
A= [1.1 2.2 3.3 4.4];
pie3(A)
title('Grafico Pie 3D')
```



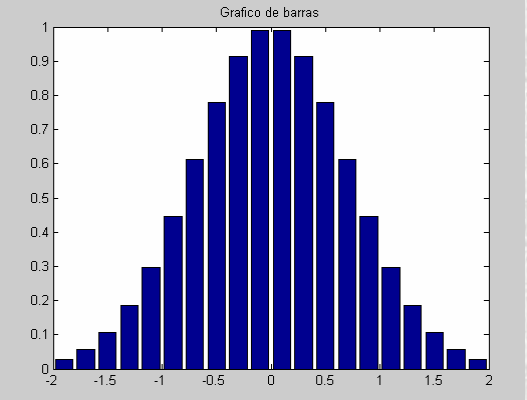
stem()

```
% gera 50 valores, 1 coluna,
distribuicao normal
% media zero, variancia 1
norma=randn(50,1)
stem(norma,'o') % mostra 50 hastes
```

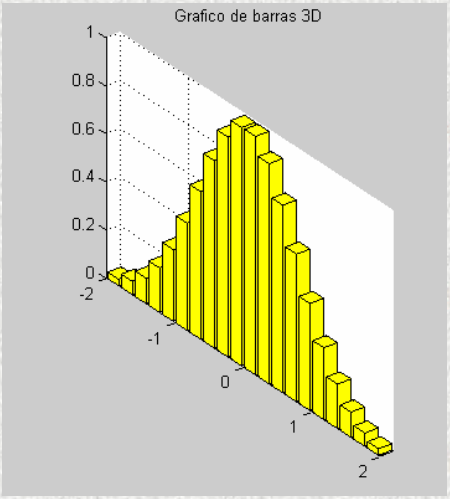


bar()

```
x=-1.9:0.2:1.9; % cria x
y=exp(-x.*x); % cria y
bar(x,y)
title('Grafico de barras')
```

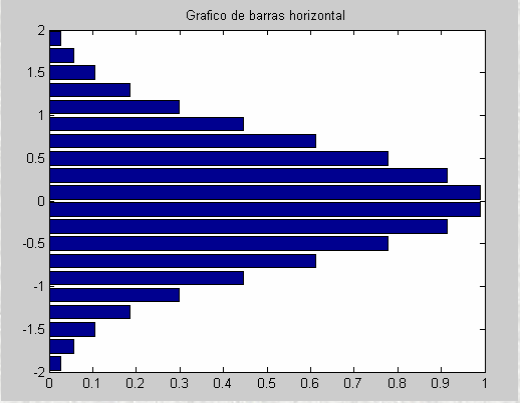


```
x=-1.9:0.2:1.9; % cria x
y=exp(-x.*x); % cria y
bar3(x,y,'y')
title('Grafico de barras 3D')
```

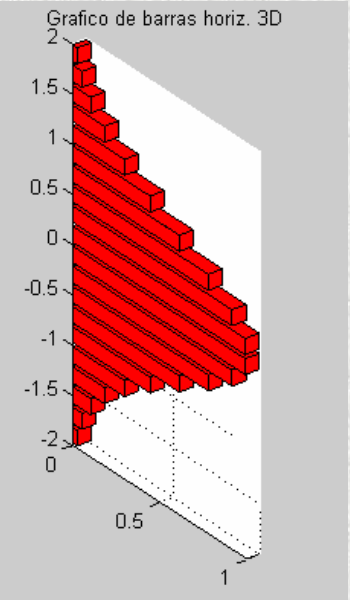


barh()

```
x=-1.9:0.2:1.9; % cria x
y=exp(-x.*x); % cria y
barh(x,y)
title('Grafico de barras horizontal')
```

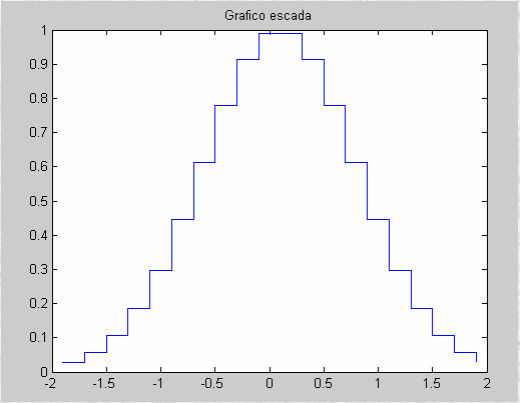


```
x=-1.9:0.2:1.9; % cria x
y=exp(-x.*x); % cria y
bar3h(x,y,'r')
title('Grafico de barras horiz. 3D')
```



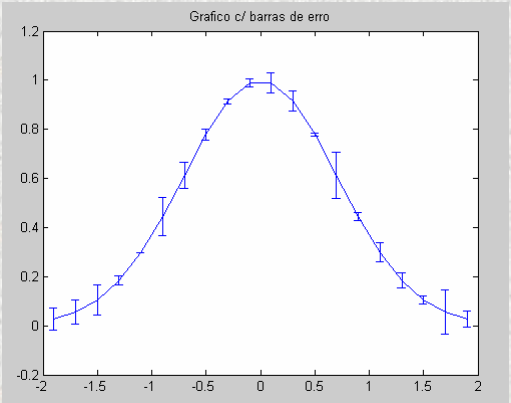
stairs()

```
x=-1.9:0.2:1.9; % cria x
y=exp(-x.*x); % cria y
stairs(x,y)
title('Grafico escada')
```

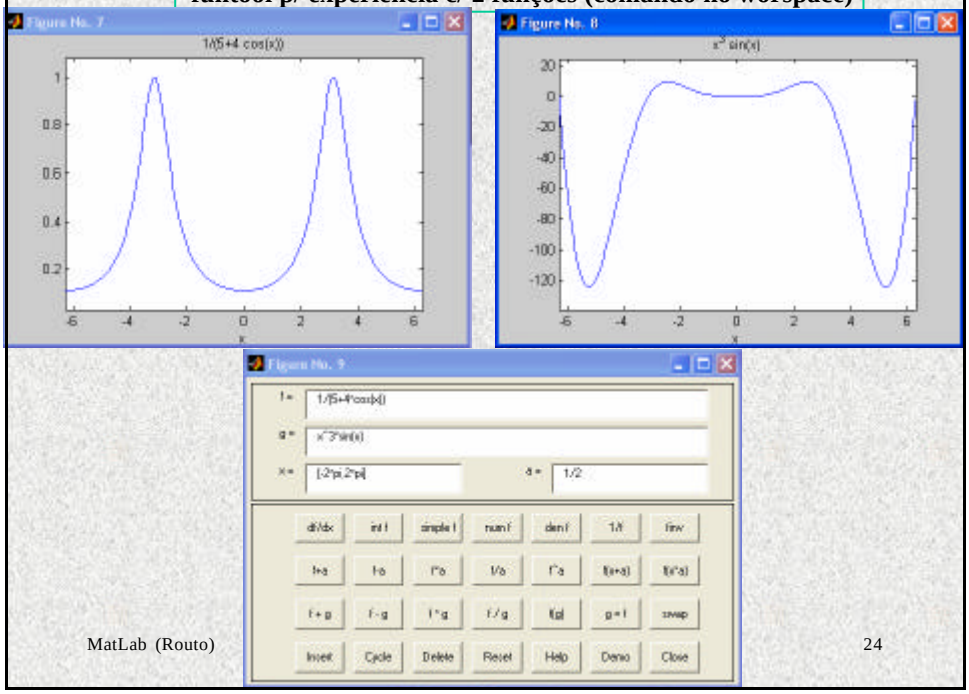


errorbar()

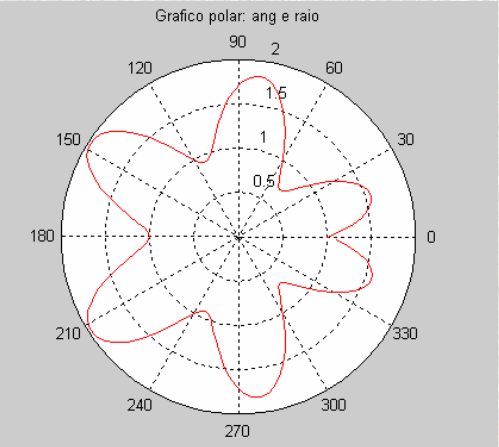
```
x=-1.9:0.2:1.9; % cria x
y=exp(-x.*x); % cria y
e=rand(size(x))/10 % pseudo aleatório
errorbar(x,y,e) % barra com y+e, y-e
title('Grafico c/ barras de erro')
```



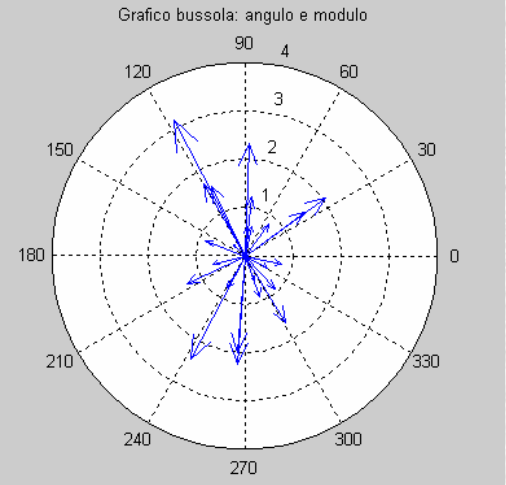
funtool p/ experiência c/ 2 funções (comando no worspace)



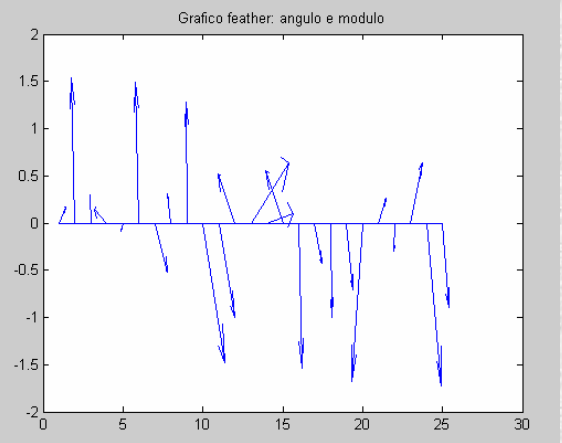
```
angulo=0:0.05:2*pi;
raio=1+sin(3*angulo).*cos(2.5*angulo);
polar(angulo,raio,'r-') % mostra ang e
raio
title('Grafico polar: ang e raio')
```



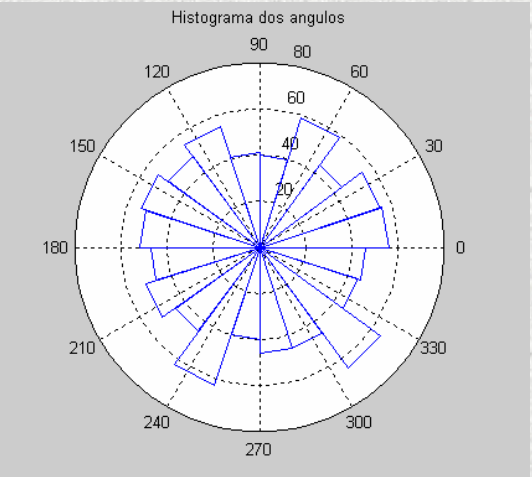

```
real=randn(5);
imag=randn(5);
compass(real,imag) % p/ numeros complexos
% mostra angulo e modulo de real+i*imag
% saindo da origem (0,0)
title('Grafico bussola: angulo e modulo')
```



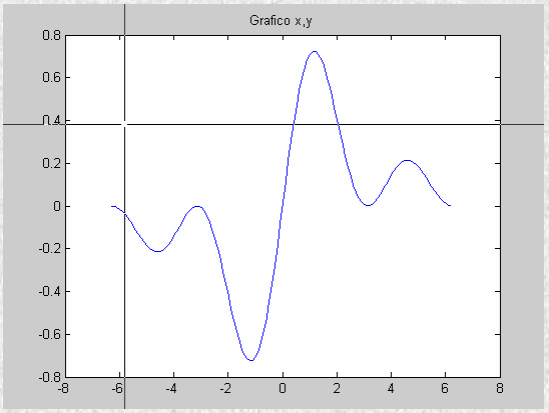
```
real=randn(5);
imag=randn(5);
feather(real,imag) % p/ numeros complexos
 % mostra angulo e modulo de real+i*imag
 % saindo da reta horizontal
title('Grafico feather: angulo e modulo')
```




```
v=randn(1000,1)*pi;
rose(v,20) % n=20
 % mostra angulos no vetor v, atraves de n "fatias"
title('Histograma dos angulos')
```

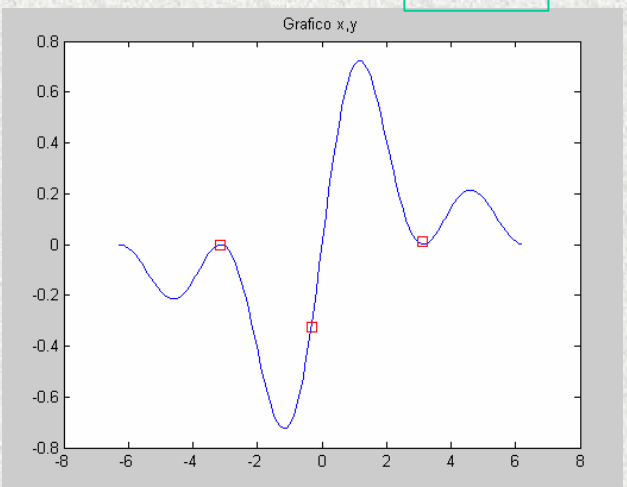


```
x=-2*pi:0.1:2*pi;
y=sin(x).^2./(x+0.001);
plot(x,y)
title('Grafico x,y')
% a seguir coleta n pontos do plano x-y atraves do mouse
[xmouse,ymouse]=ginput(3)
hold on
plot(xmouse,ymouse,'rs') % red square
hold off
```



Ptos. coletados pelo mouse

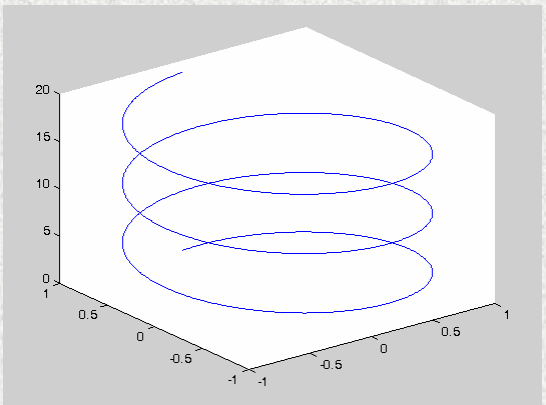
```
xmouse =
   -3.1521
   -0.3134
    3.1152
ymouse =
   -0.0023
   -0.3251
    0.0117
```



Gráficos 3D

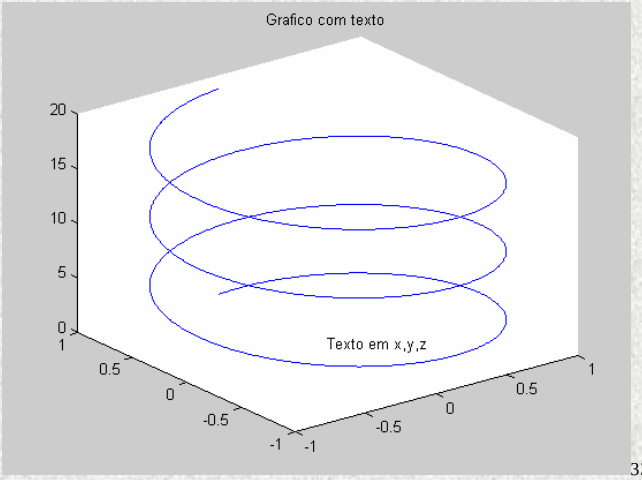
plot3()

```
% plot3, helice (sen(t),cos(t),t)
t=0:0.01:6*pi; % intervalo para eixo t
plot3(sin(t), cos(t), t) % eixos x,y,z
```



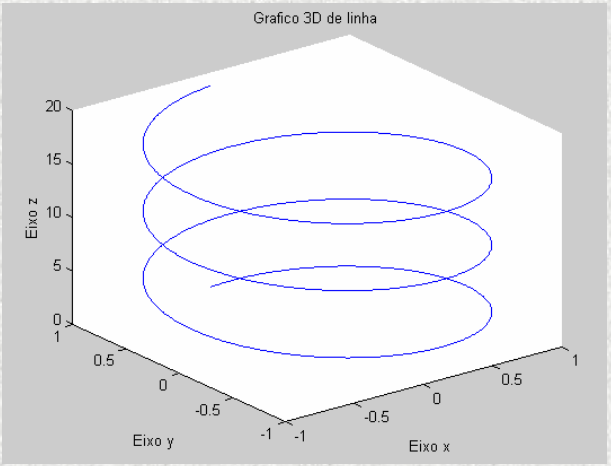
plot3()

```
t=0:0.01:6*pi; % intervalo para eixo t
plot3(sin(t), cos(t), t) % eixos x,y,z
title('Grafico com texto')
text(0,0,0,'Texto em x,y,z')
```



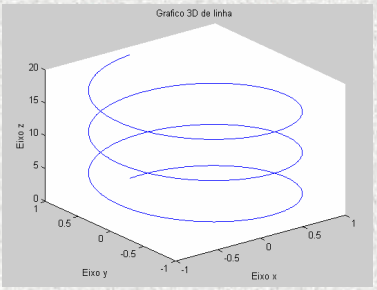
plot3()

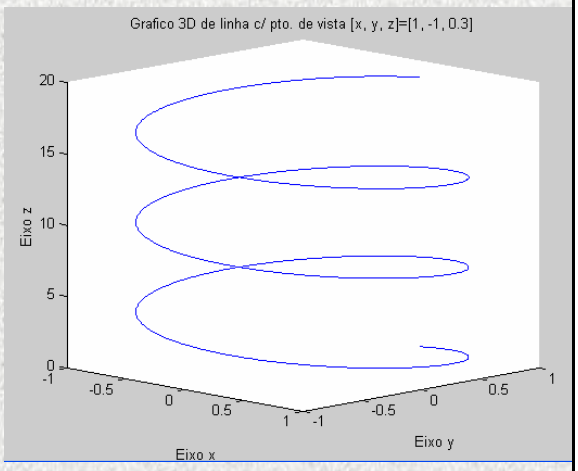
```
t=0:0.01:6*pi; %define eixo z
plot3(sin(t),cos(t),t) % eixos x, y, z
xlabel('Eixo x')
ylabel('Eixo y')
zlabel('Eixo z')
title('Grafico 3D de linha')
```



```
t=0:0.01:6*pi;
plot3(sin(t),cos(t),t)
xlabel('Eixo x')
ylabel('Eixo y')
zlabel('Eixo z')
title('Grafico 3D de linha c/ pto. de vista [x, y, z]=[1, -1, 0.3]')
% a seguir, define ponto de vista 3D
view([1, -1, 0.3])  % note [x, y, z]
        % view([-1,-1,1]) e' o default
```

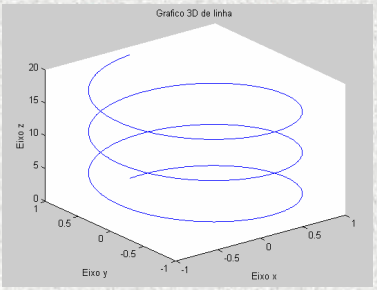
view() p/ definir ponto de vista 3D

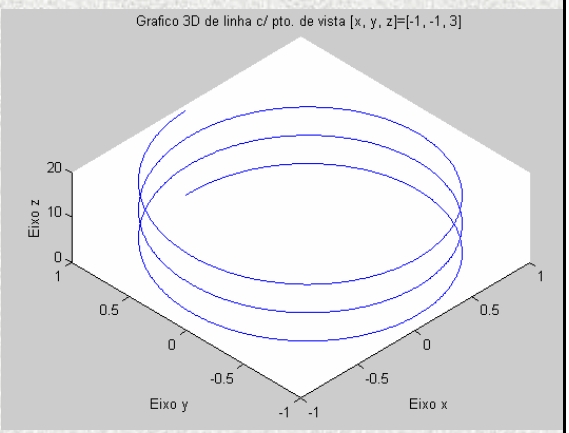





```
t=0:0.01:6*pi;
plot3(sin(t),cos(t),t)
xlabel('Eixo x')
ylabel('Eixo y')
zlabel('Eixo z')
title('Grafico 3D de linha c/ pto. de vista [x, y, z]=[-1, -1, 3]')
% a seguir, define ponto de vista 3D
view([-1, -1, 3])  % note [x, y, z]
        % view([-1,-1,1]) e' o default
```

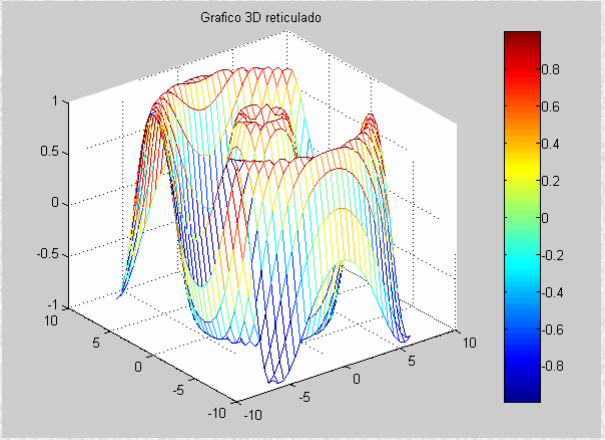
view() com [-1,-1,3]





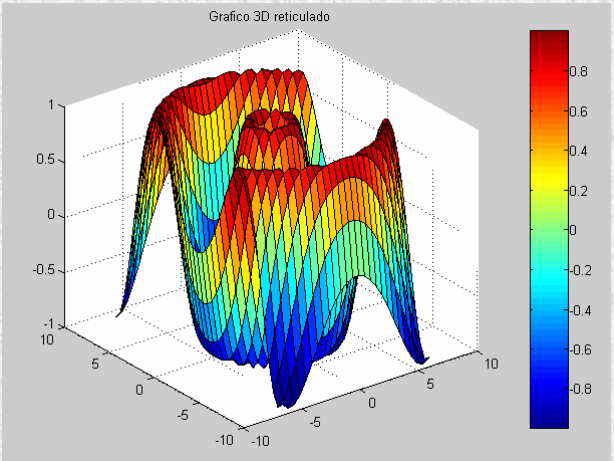
mesh() c/ 3 argu-mentos

```
% definir uma "grade" de pontos X Y no plano x-y
% aplicando meshgrid()
[X, Y]= meshgrid(-2*pi:0.5:2*pi, -3*pi:0.5:3*pi); % note o ;
% a seguir grafico 3D reticulado
% 3o. argumento e' o valor no eixo Z
mesh(X,Y,sin(sqrt(X.*X+Y.*Y)))
title('Grafico 3D reticulado')
colorbar  % a cor indica o valor no eixo Z
```



surf()

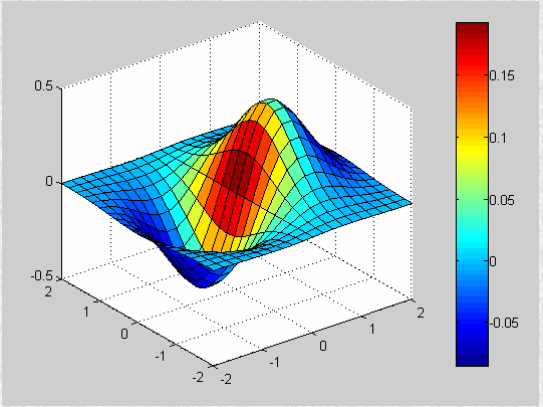
```
% definir uma "grade" de pontos X Y no plano x-y
% aplicando meshgrid()
[X, Y]= meshgrid(-2*pi:0.5:2*pi, -3*pi:0.5:3*pi); % note o ;
% a seguir grafico 3D com "retalhos"
% preenchendo cada reticulado
% 3o. argumento e' o valor no eixo Z
surf(X,Y,sin(sqrt(X.*X+Y.*Y)))
title('Grafico 3D reticulado c/ preenchimento')
colorbar  % a cor indica o valor no eixo Z
```



surf()
c/ 4o. argumento

O 4o. argumento indica a cor aplicada.
Gradiente é a inclinação/derivada.

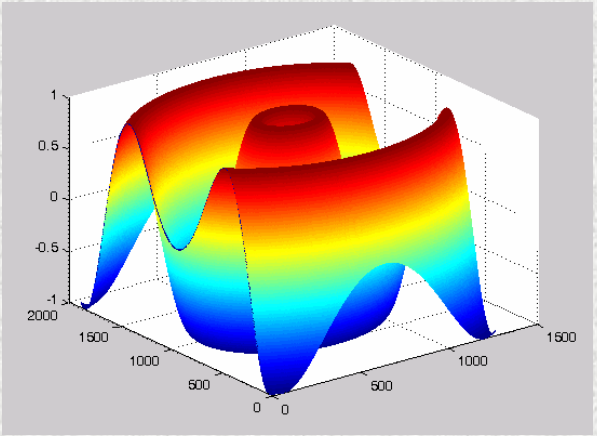
```
[x,y]= meshgrid([-2:.2:2]);
 Z= x.*exp(-x.^2-y.^2);
surf(x,y,Z,gradient(Z))
 colorbar
```



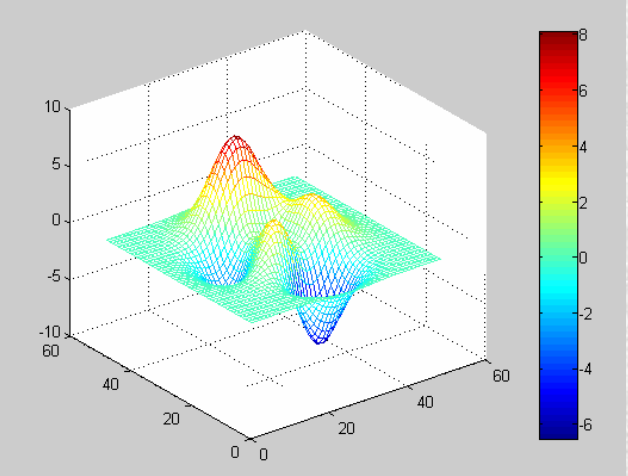
mesh()
c/ 1 argumento

Aviso: geração deste gráfico demora muito!

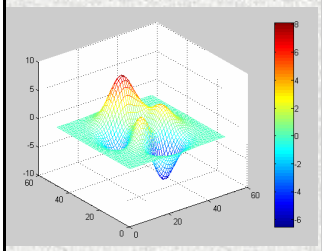
```
% a seguir definir um dominio X Y
[X, Y]= meshgrid( -2*pi:0.01:2*pi, -3*pi:0.01:3*pi); % note o ;
% índice de X no eixo x, índice de Y no eixo y
mesh( sin(sqrt(X.*X+Y.*Y)) ) % só 1 argumento, p/ o eixo Z
```

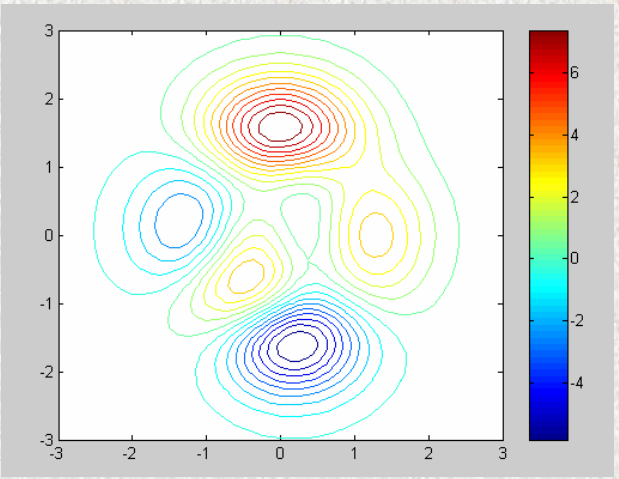


mesh(peaks) % função peaks() pré-definida
colorbar

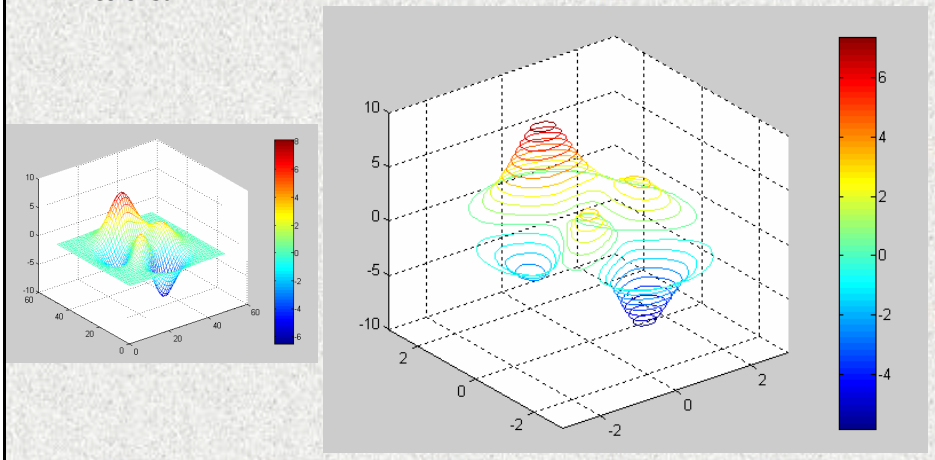


mesh(peaks) % função peaks() pré-definida
[x,y,z]=peaks;
contour(x,y,z,20) % 20 curvas de nivel
colorbar

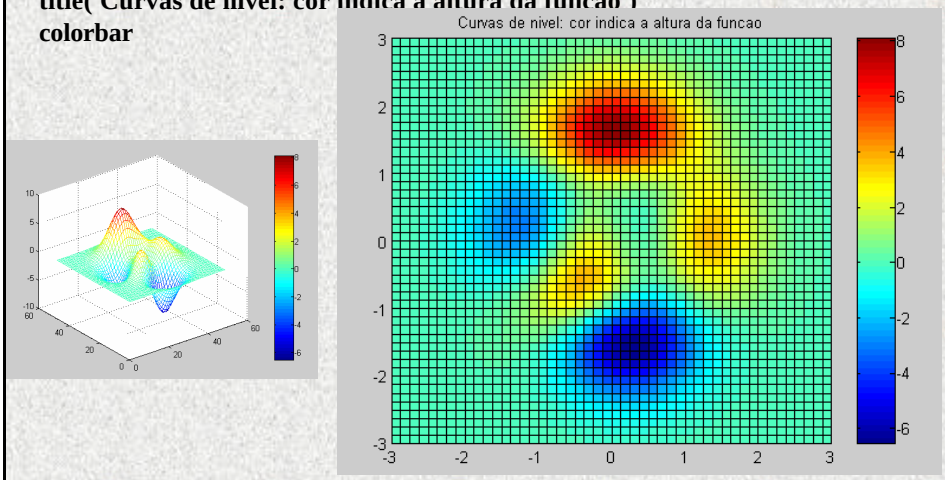




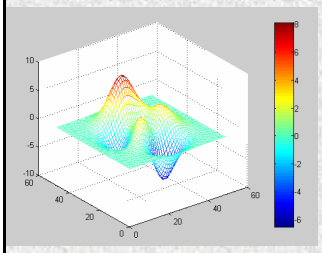
mesh(peaks) % função peaks() pré-definida
[x,y,z]=peaks;
contour3(x,y,z,20) % 20 curvas de nivel 3D
colorbar

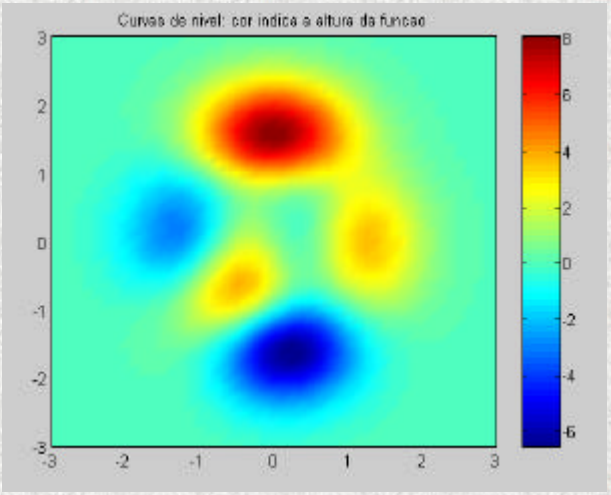


mesh(peaks) % função peaks() pré-definida
[x,y,z]=peaks;
pcolor(x,y,z) % cor indica a altura da função
title('Curvas de nivel: cor indica a altura da funcao')
colorbar

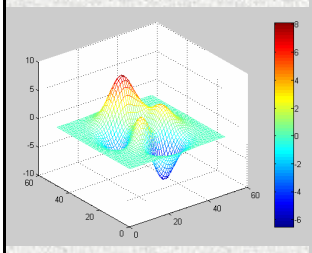


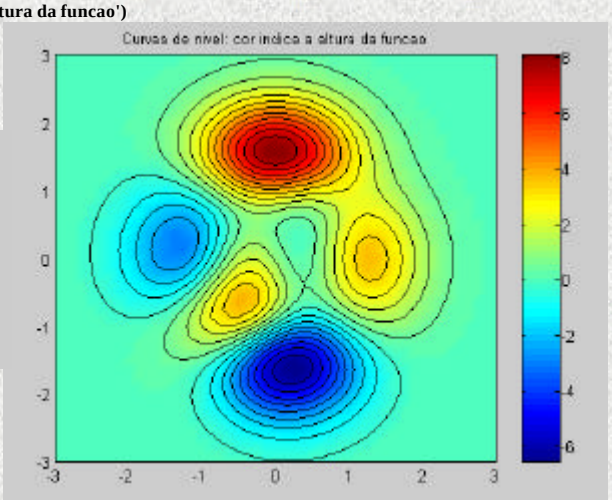
```
mesh(peaks) % função peaks() pré-definida
[x,y,z]=peaks;
pcolor(x,y,z) % cor indica a altura da função
shading interp % remove o reticulado de linhas
title('Curvas de nivel: cor indica a altura da funcao')
colorbar
```



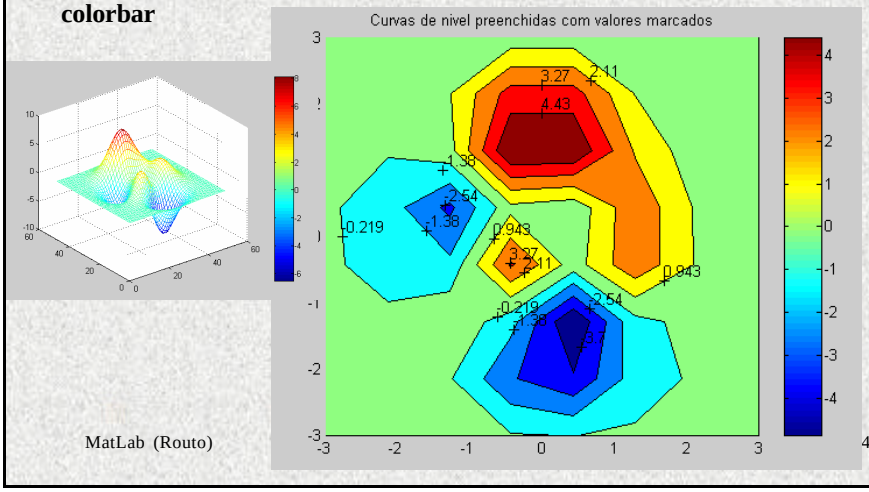



```
mesh(peaks) % função peaks() pré-definida
[x,y,z]=peaks;
pcolor(x,y,z) % cor indica a altura da função
shading interp % remove o reticulado de linhas
hold on
% superpor 20 curvas de linha em cor preta
contour(x,y,z,20,'k')
title('Curvas de nivel: cor indica a altura da funcao')
colorbar
hold off
```



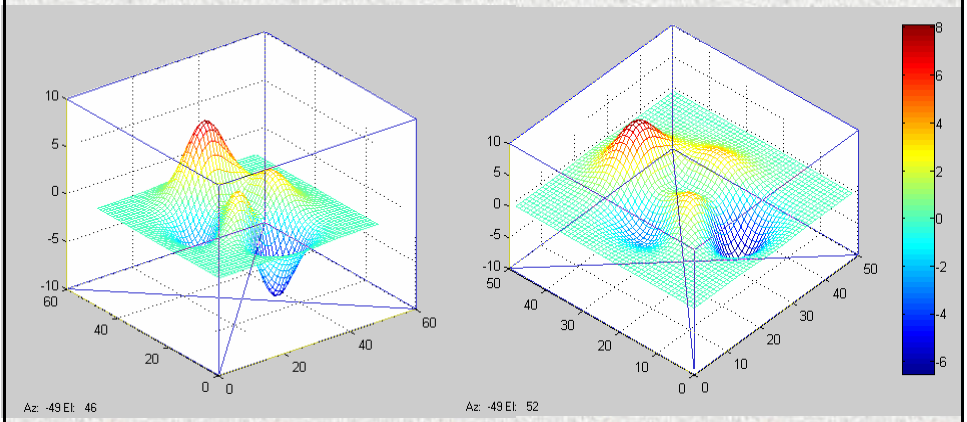


```
N=8
[x,y,z]=peaks(N);
% N curvas de nivel preenchidas: contourf()
curva=contourf(x,y,z,N);
% clabel() para mostrar valores
clabel(curva)
title('Curvas de nivel preenchidas com valores marcados')
colorbar
```

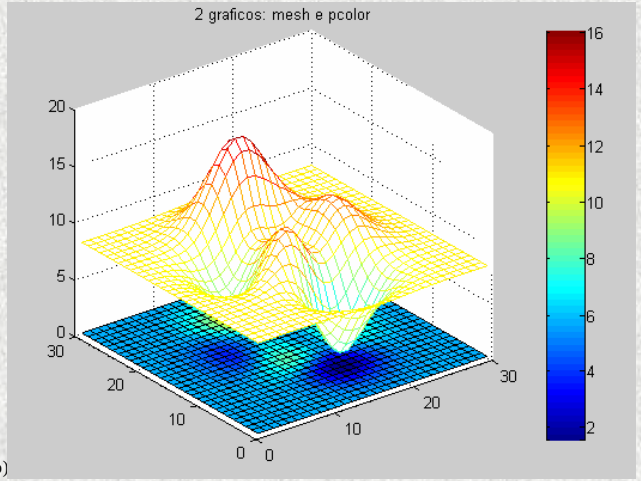



```
mesh(peaks)
colorbar
rotate3d on
```

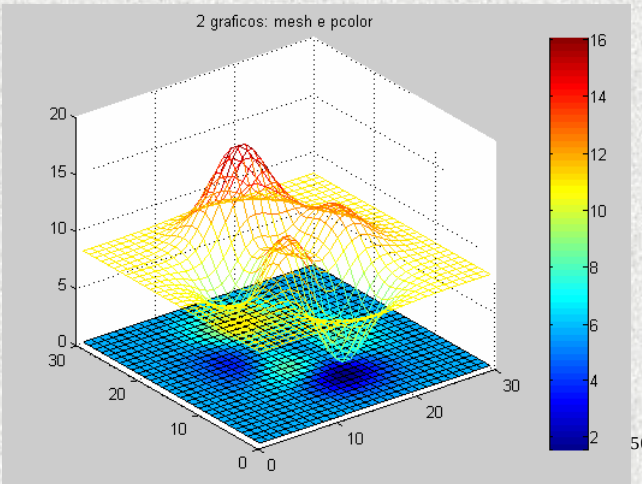
rotate3d on p/ ligar a opção de arrastar o gráfico com o mouse e mudar o pto. de vista 3D



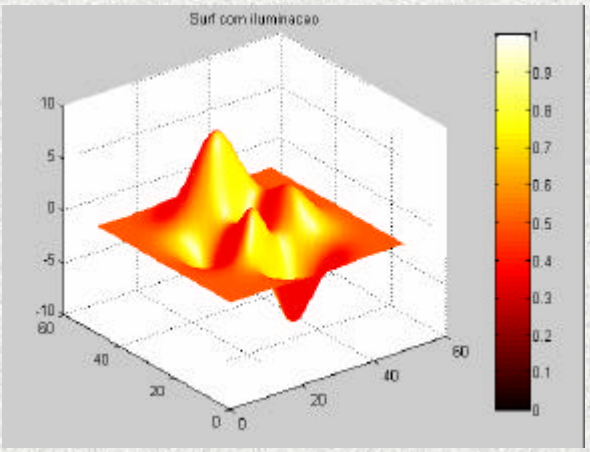
```
mesh(peaks(30)+8) % peaks com 30 fatias
colorbar
hold on
pcolor(peaks(30))
hold off
title('2 graficos: mesh e pcolor')
```




```
mesh(peaks(30)+8) % peaks com 30 fatias
colorbar
hold on
pcolor(peaks(30))
hold off
title('2 graficos: mesh e pcolor')
hidden off % p/ mostrar as linhas "escondidas"
```



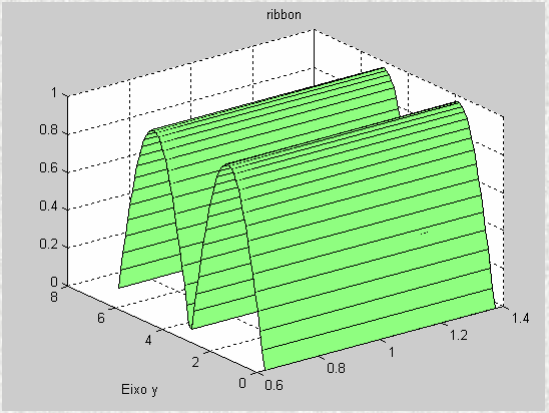
colormap(hot) % escolhe mapa de cores
surfl(peaks) % surfl p/ iluminar com constrate
shading interp; % tira reticulado
title('Surf com iluminacao')
colorbar



Mapas de cores

hsv	Cores saturadas
hot	Preto+vermelho+amarelo+branco
gray	Tons de cinza
bone	Cinza azulada
copper	cobre
pink	rosa
white	
flag	Verm+branco+azul+preto alternados
jet	Parece hsv
prism	
cool	Ciano e magenta
lines	Cores da função plot()
colorcube	
summer	Amarelo+verde
autumn	Verm+amarelo
winter	Azul+verde
spring	Magenta +amarelo

```
x=0:0.1:2*pi;
y=abs(sin(x));
ribbon(x,y) % y como fitas
ylabel('Eixo y')
title('ribbon')
```




```
N=22;
meshz(peaks(N));
title('meshz p/ grafico "cortina"')
```