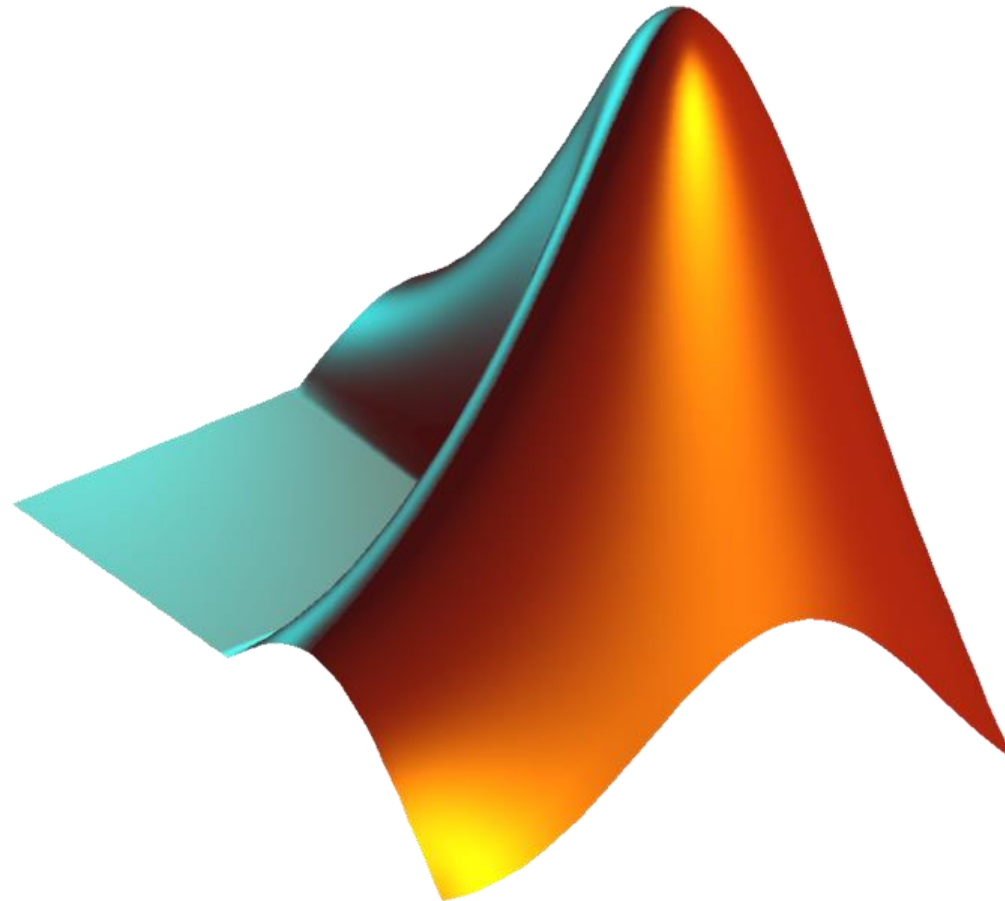




MATLAB COURSE



مدرس دوره : احمد خیراندیش

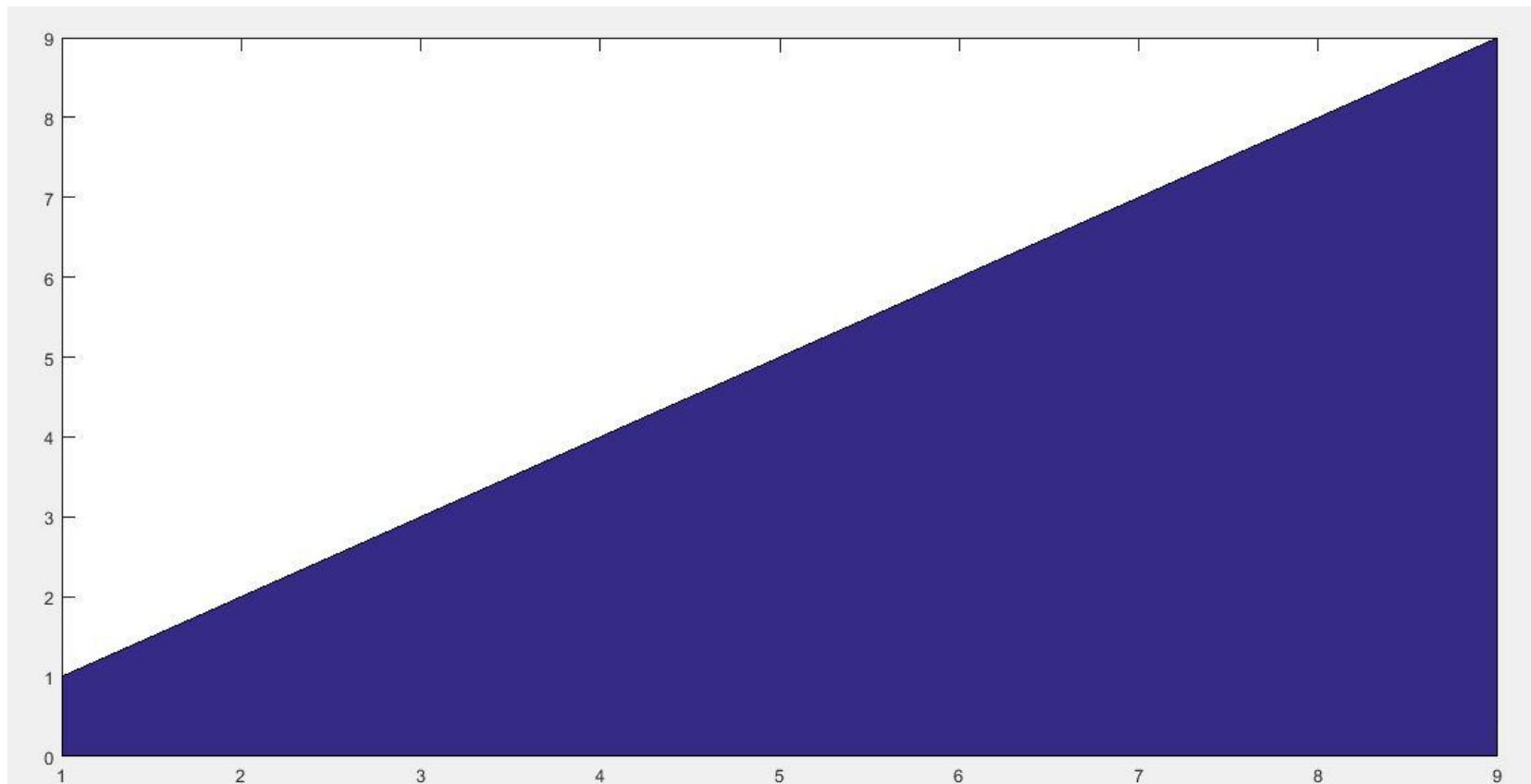
Session: **2**

ترسیمات دو بعدی خاص

area	ایجاد یک رسم ناحیه ای پر شده
bar	ایجاد نمودار میله ای
barh	ایجاد نمودار میله ای افقی
compass	ایجاد گراف های جهت دار برای اعداد مختلط
feather	ایجاد یک نمودار فیدر
hist	ایجاد هیستوگرام
pcolor	ایجاد نمودارهای رنگی کاذب از یک ماتریس
pie	ایجاد چارت های دایره ای
polar	رسم منحنی در مختصات قطبی
stairs	رسم یک گراف پلکانی
stem	رسم یک گراف شاخه ای

area

```
y = [1, 2, 3, 4, 5, 6, 7, 8, 9];  
area(y)
```



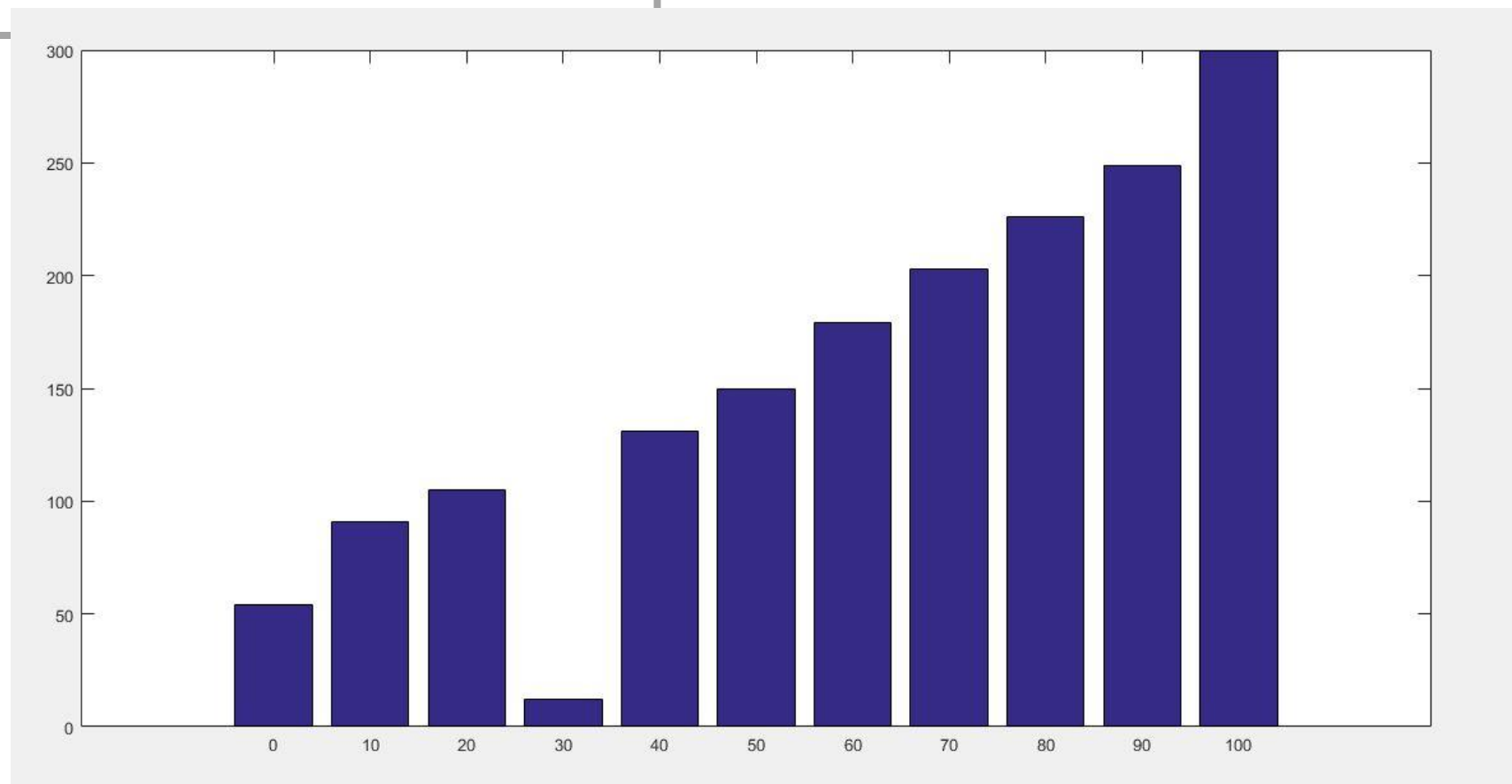
bar

مثال های ترسیمات دو بعدی خاص

```
x = 0:10:100;
```

```
y = [54 91 105 12 131 150 179 203 226 249 300];
```

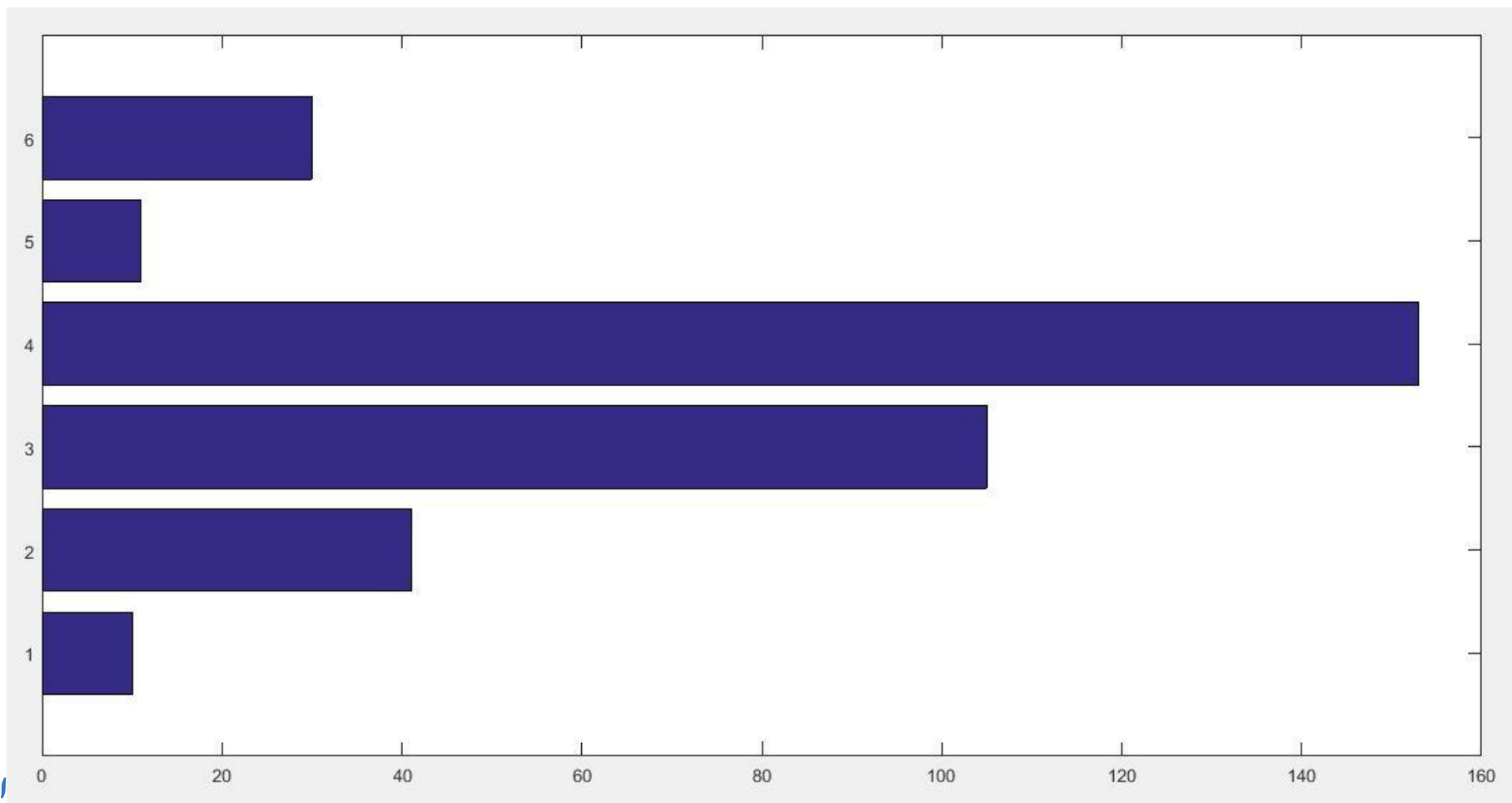
```
bar(x,y)
```



مثال های ترسیمات دو بعدی خاص

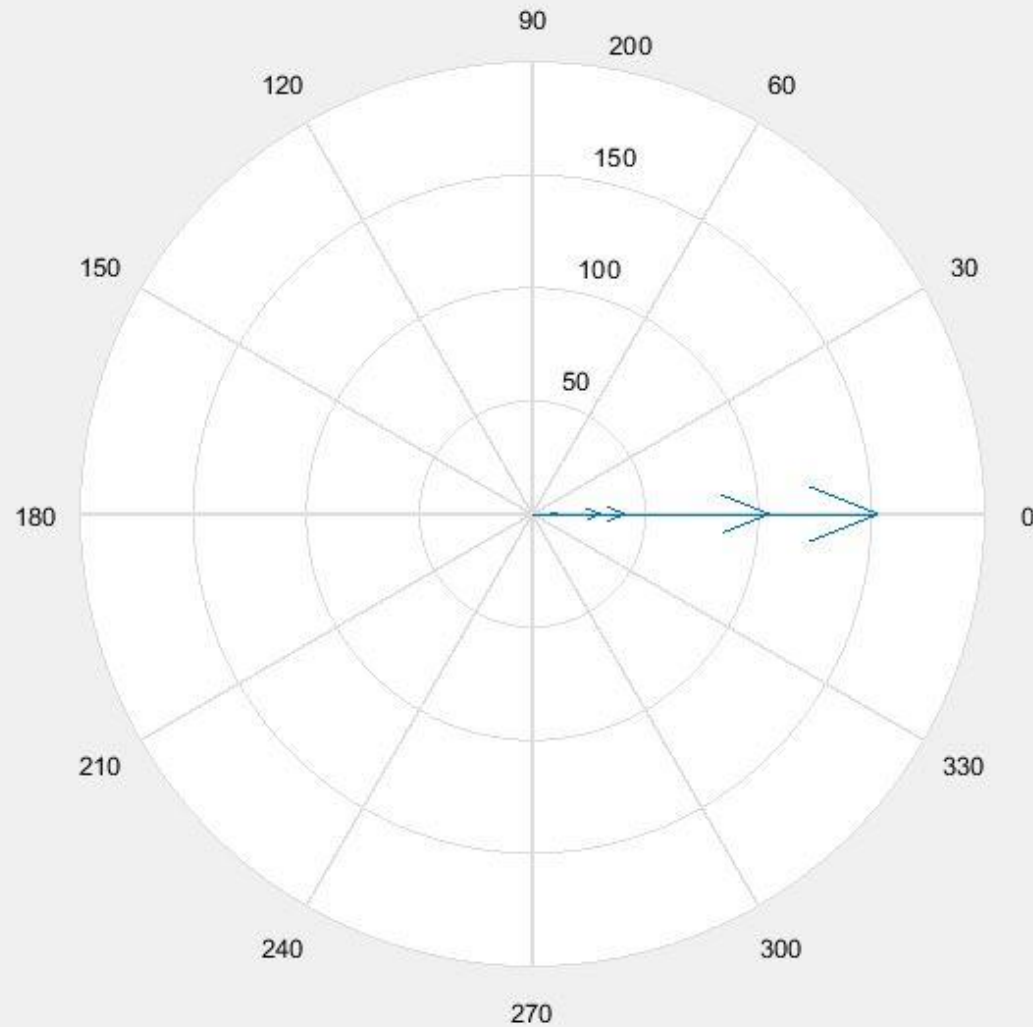
barh

```
y = [10,41,105,153,11,30];  
barh(y)
```



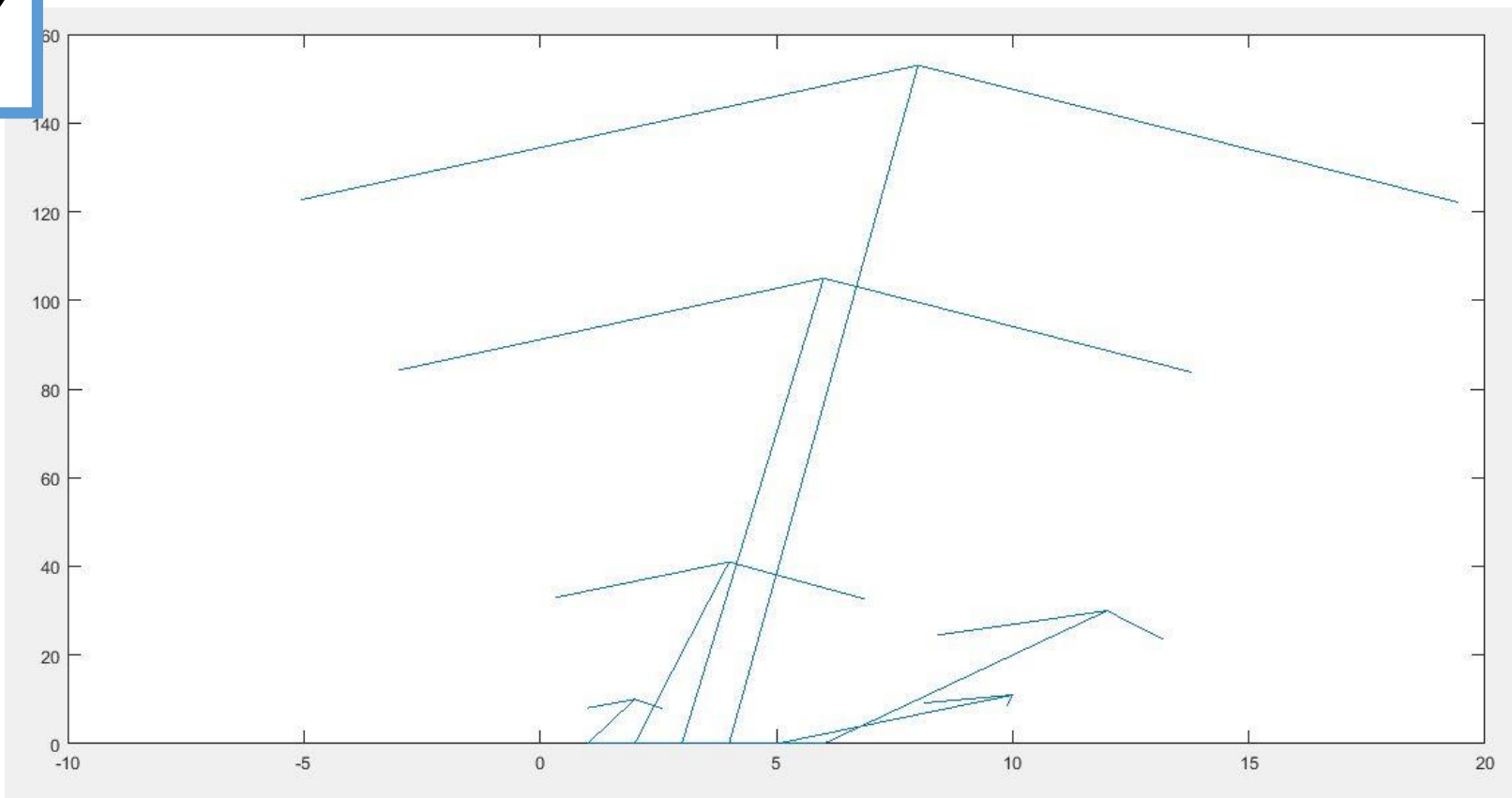
compass

```
y = [10,41,105,153,11,30 ];  
compass(y)
```



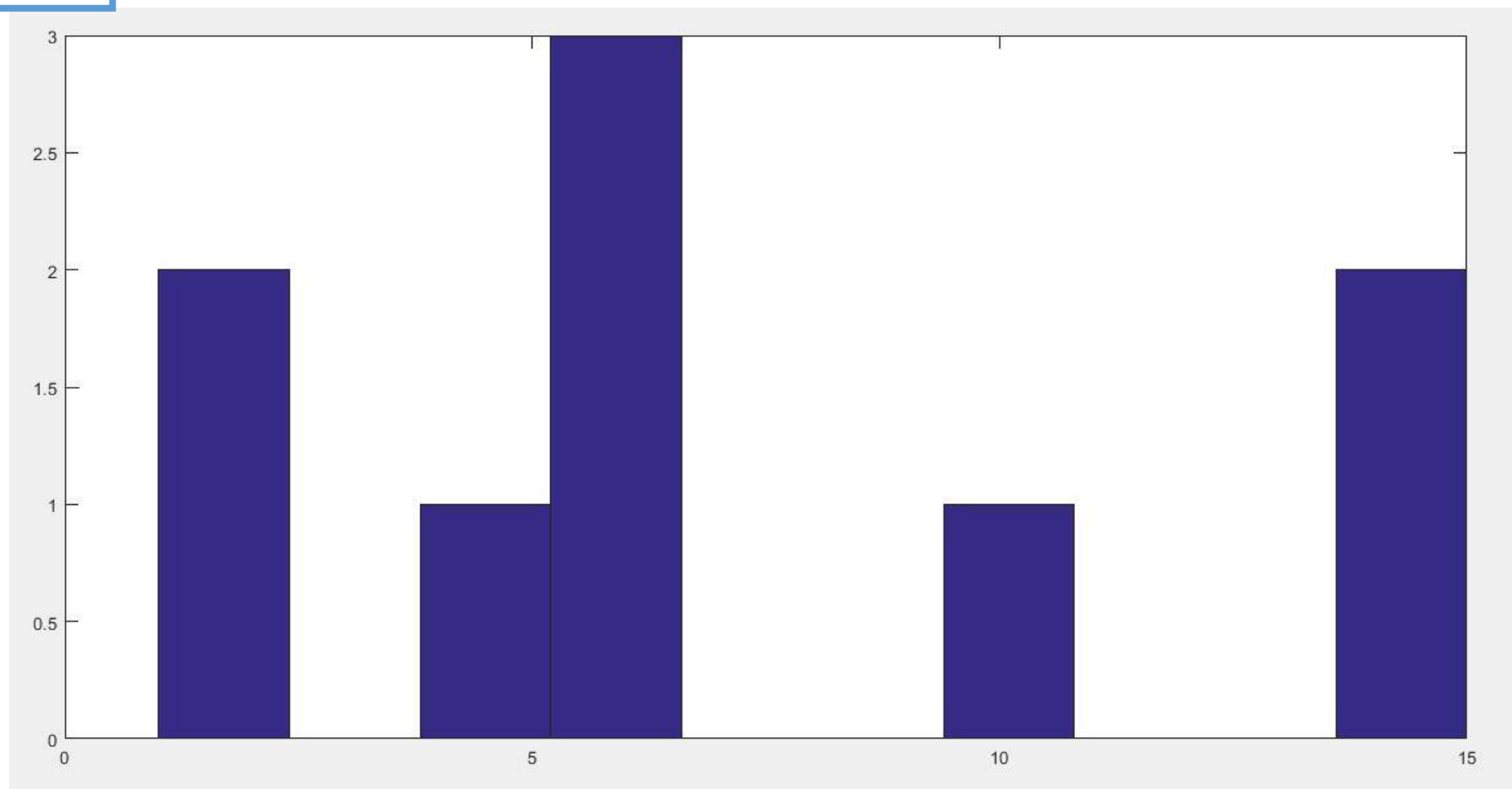
feather

```
x = [1,2,3,4,5,6 ];  
y = [10,41,105,153,11,30 ];  
feather(x,y)
```



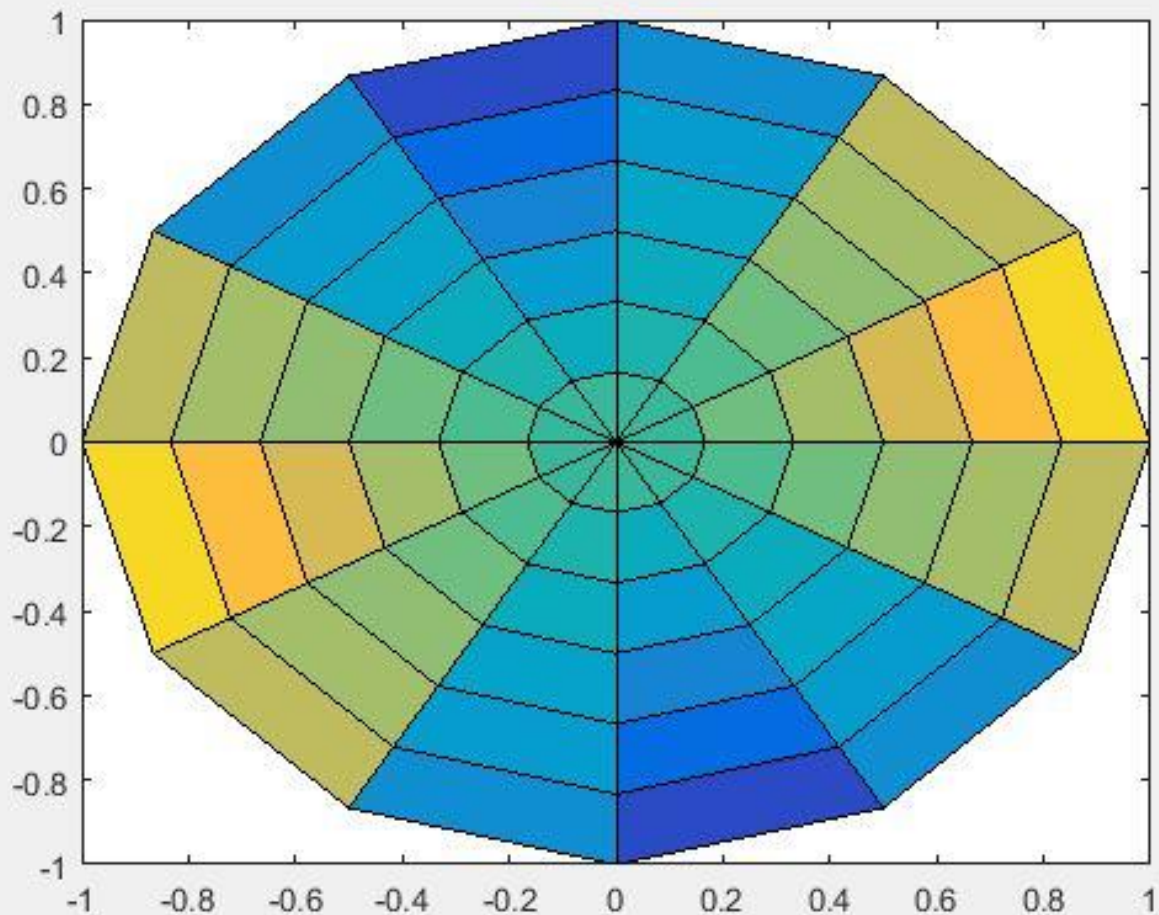
hist

```
x=[1,5,6,6,6,1,10,15,15];  
hist(x)
```



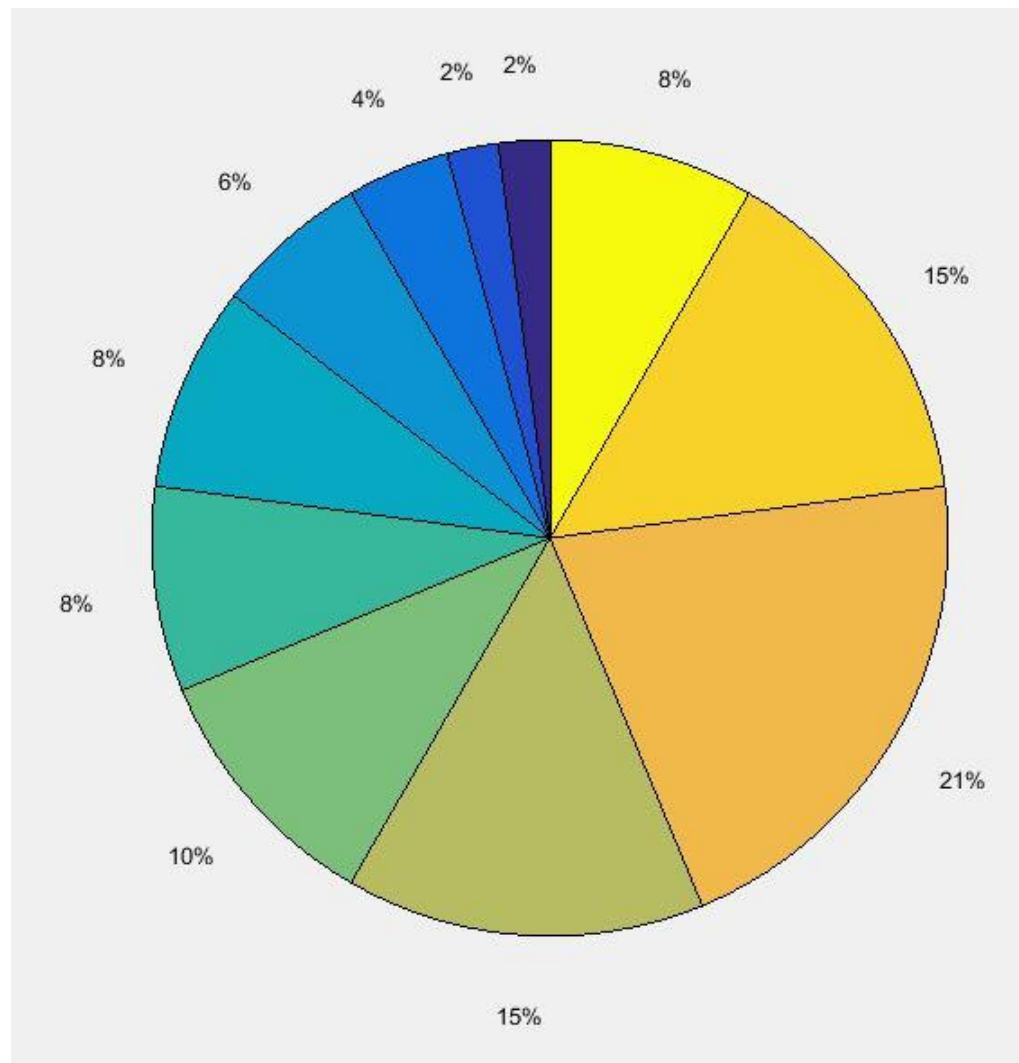
pcolor

```
n = 6;  
r = (0:n)'/n;  
theta = pi*(-n:n)/n;  
x = r*cos(theta);  
y = r*sin(theta);  
c = r*cos(2*theta);  
pcolor(x,y,c)
```



pie

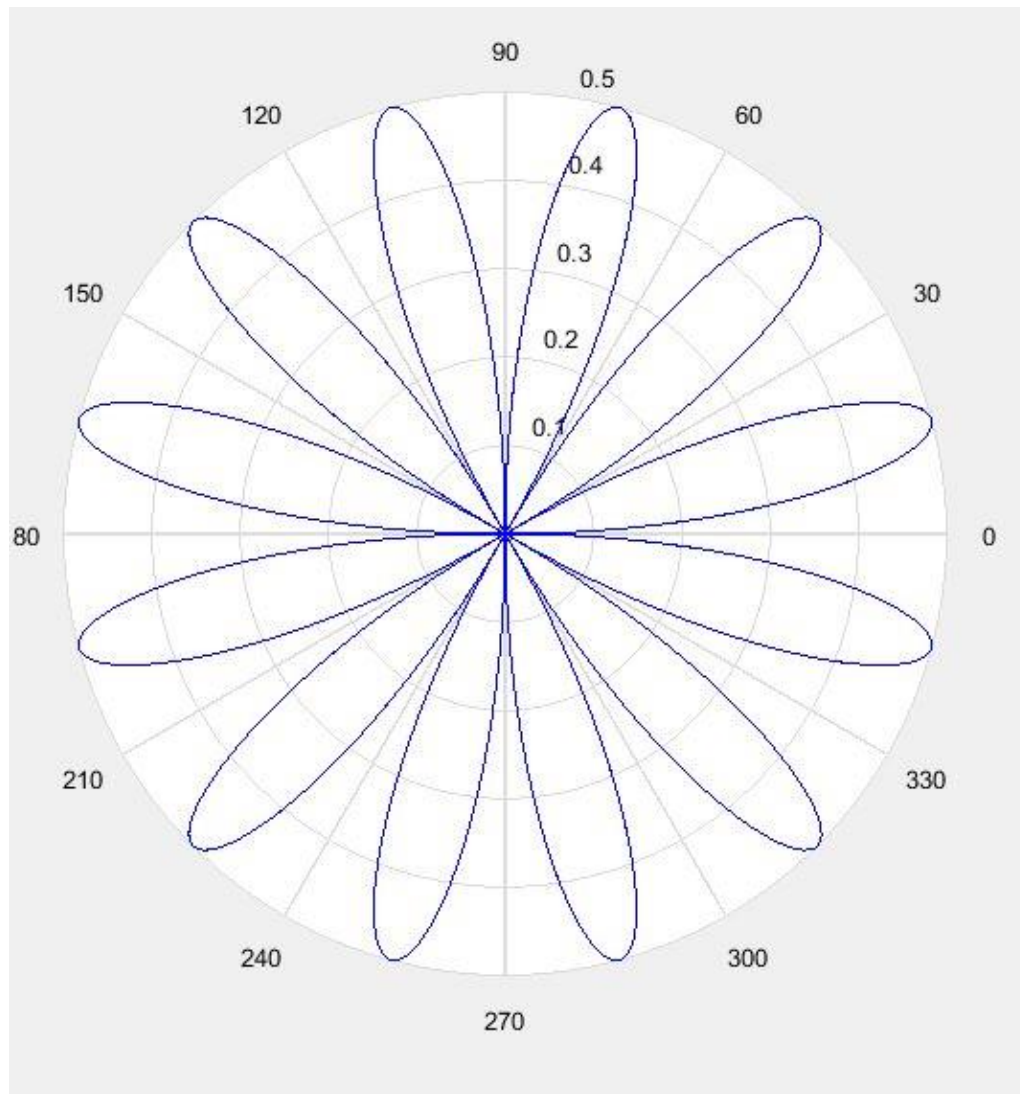
```
x = [1 1 2 3 4 4 5 7 10 7 4];  
pie(x)
```



مثال های ترسیمات دو بعدی خاص

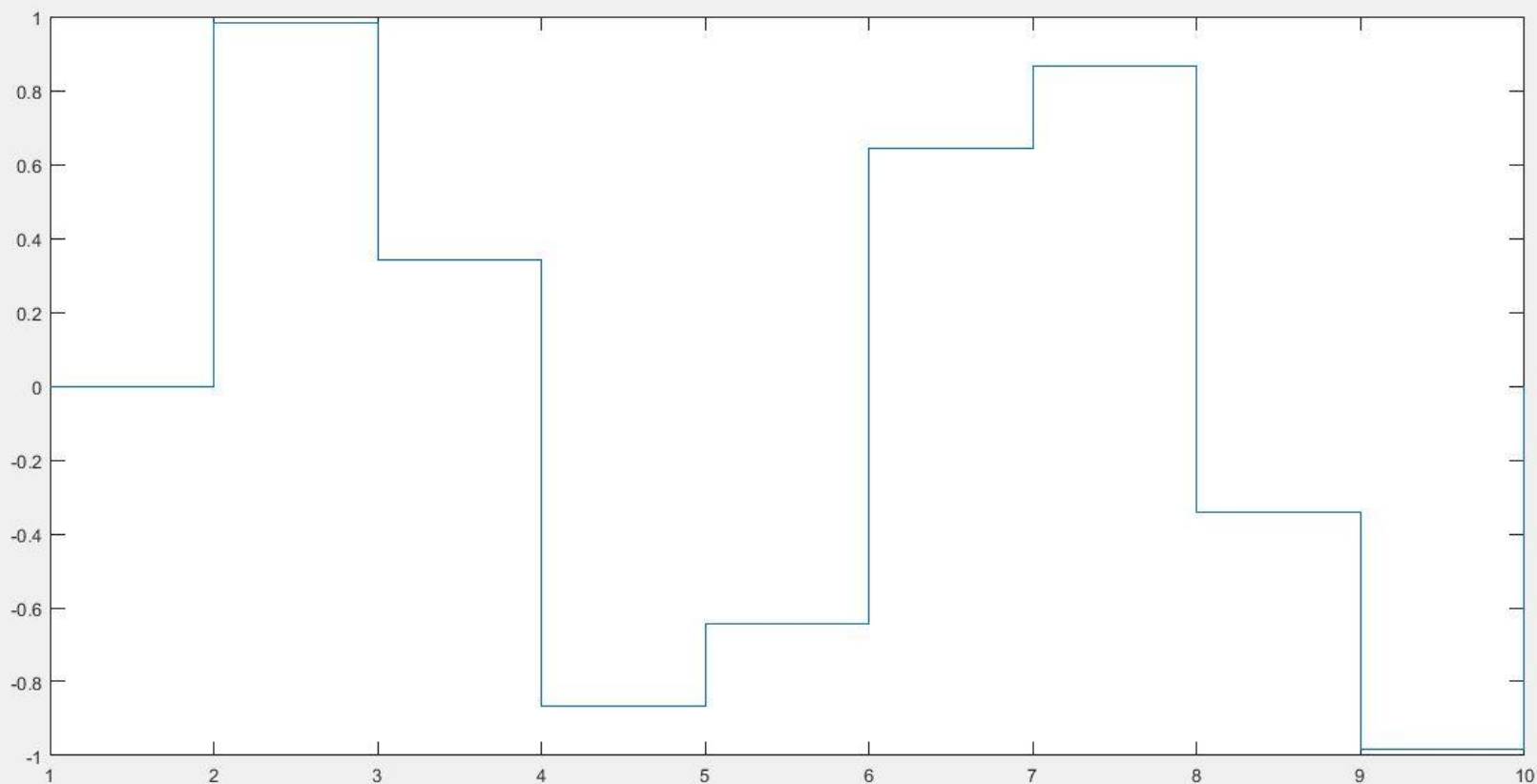
polar

```
theta = 0:0.01:3*pi;  
rho = sin(3*theta).*cos(3*theta);  
polar(theta,rho,'-b')
```



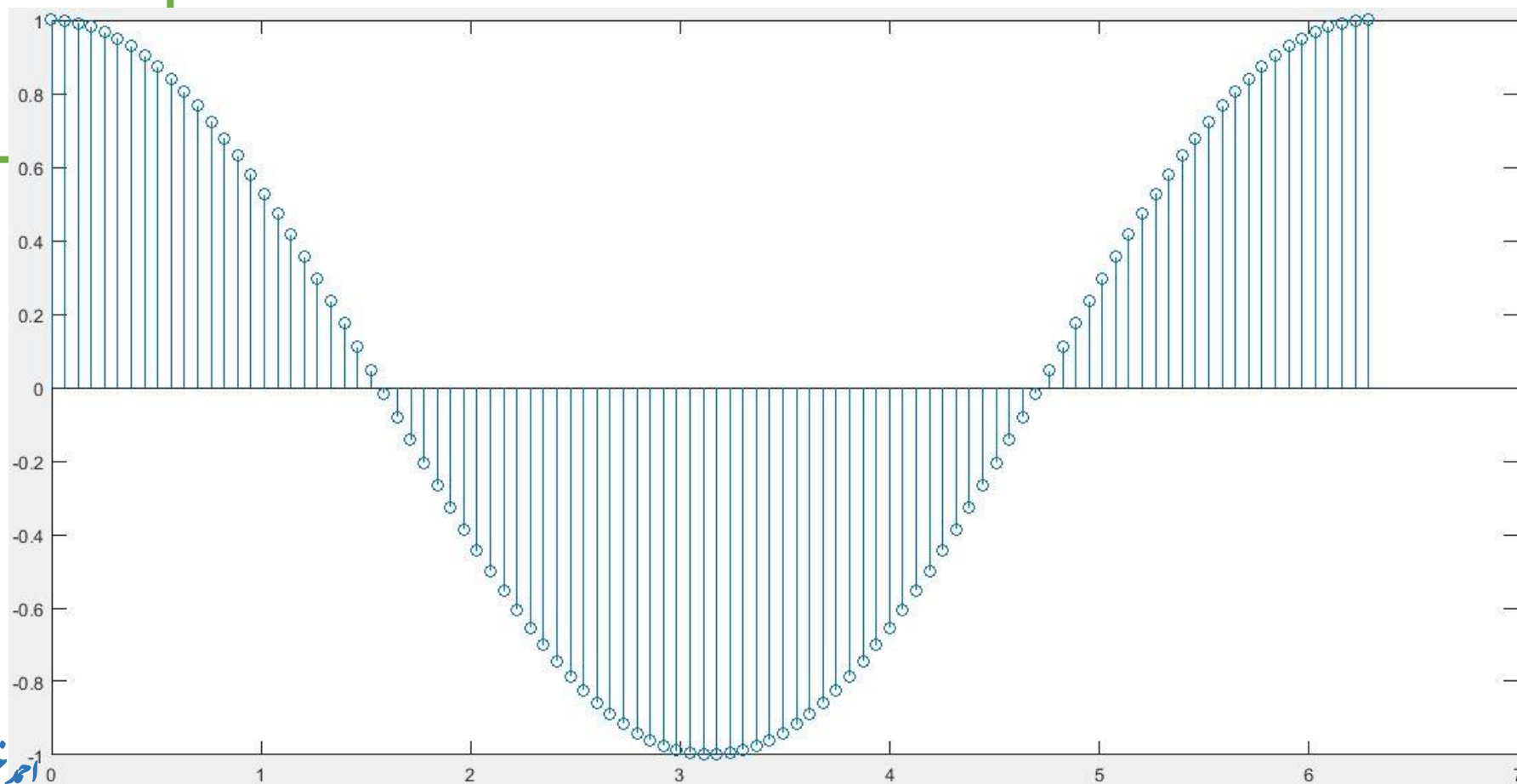
stairs

```
x = linspace(0,4*pi,10);  
y = sin(x);  
stairs(y)
```



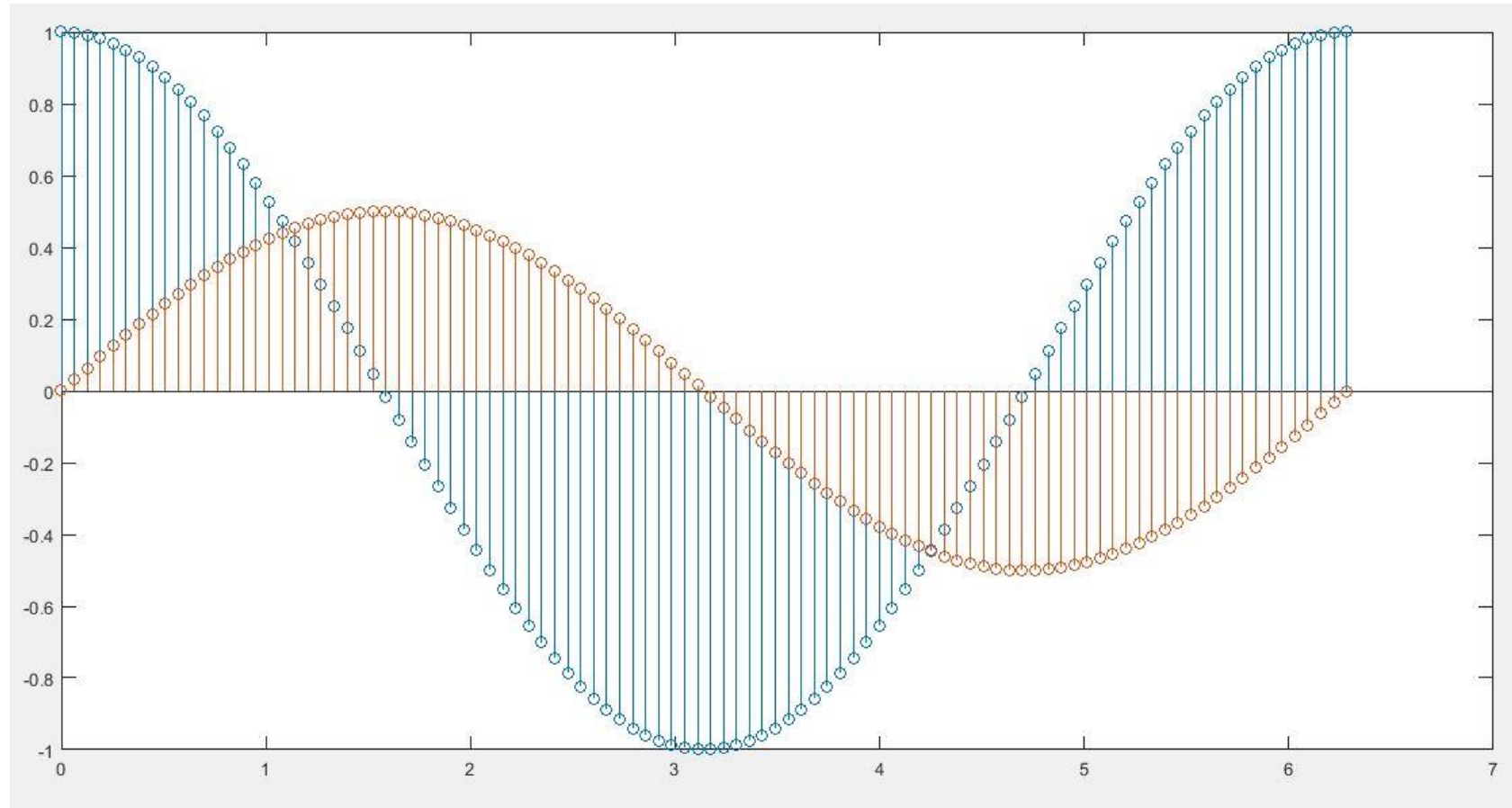
stem

```
x = linspace(0,2*pi,100)';  
y = cos(x);  
stem(x,y)
```



stem

```
x = linspace(0,2*pi,100)';  
y = [cos(x),0.5*sin(x)];  
stem(x,y)
```



دستورات ترسیمات سه بعدی

اگر x, y دو بردار شامل آرایه هایی از نقاطی باشند، این دستور دو ماتریس مستطیلی را بر می گرداند که شامل مقادیر x, y در هر نقطه از شبکه های ۲ بعدی است.

meshgrid

$[X,Y]=\text{meshgrid}(x,y)$

$[X,Y]=\text{meshgrid}(1:0.5:10, 0:2:20)$

دستورات ترسیمات سه بعدی

mesh

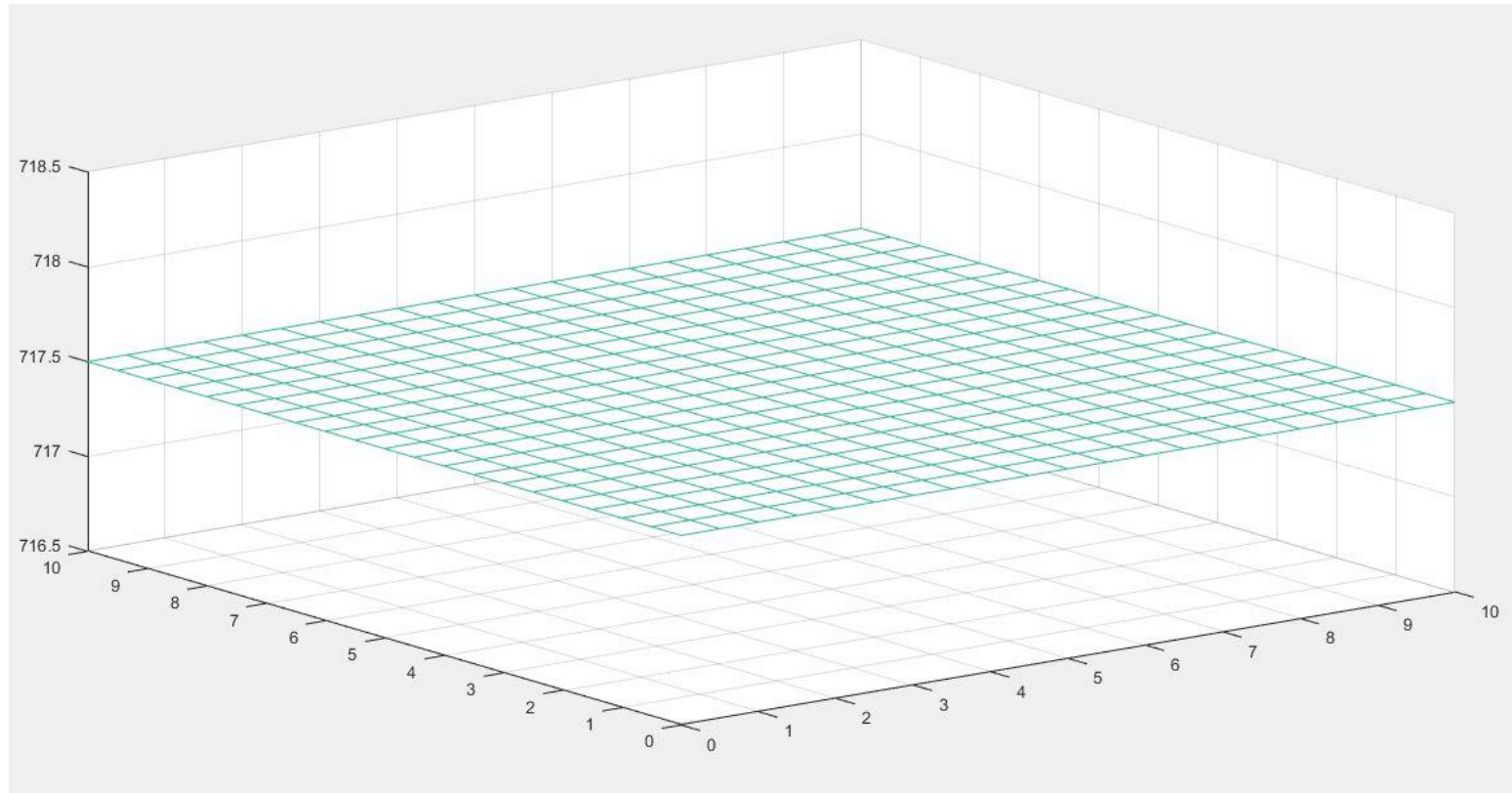
اگر X, Y آرایه های یک شبکه مستطیلی باشند و اگر Z مقدار یک تابع ارزیابی شده در هریک از این نقاط باشد آنگاه این دستور، یک نمودار سه بعدی از نقاط را رسم خواهد کرد.

`mesh (X , Y , z)`

```
[X,Y] = meshgrid(-12:0.7:18);  
R = sqrt(X.^2 + Y.^2) + eps;  
Z = sin(R)./R;  
mesh(Z)
```

دستورات ترسیمات سه بعدی

```
[X,Y] = meshgrid(0: .5 :10);  
R =( X*Y) ;  
mesh(X,Y,R)
```

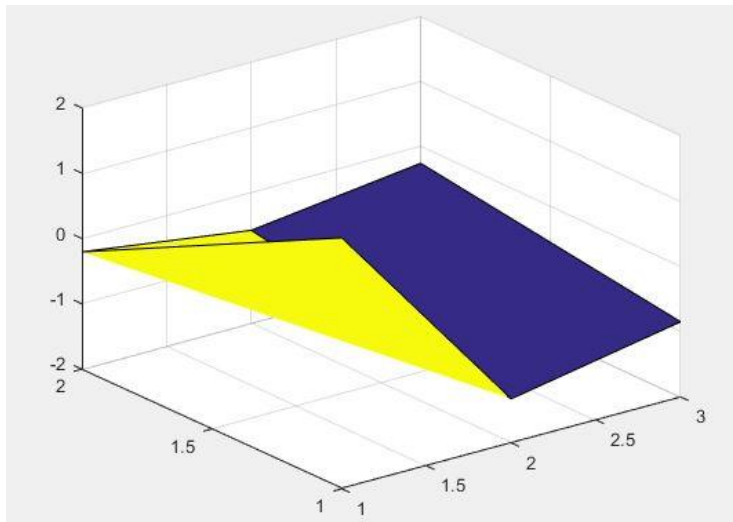


surf

این دستور یک نمای سه بعدی را رسم می کند و معمولاً برای رسم سطح استفاده می شود. خروجی دستور، یک شکل سایه دار خواهد بود.

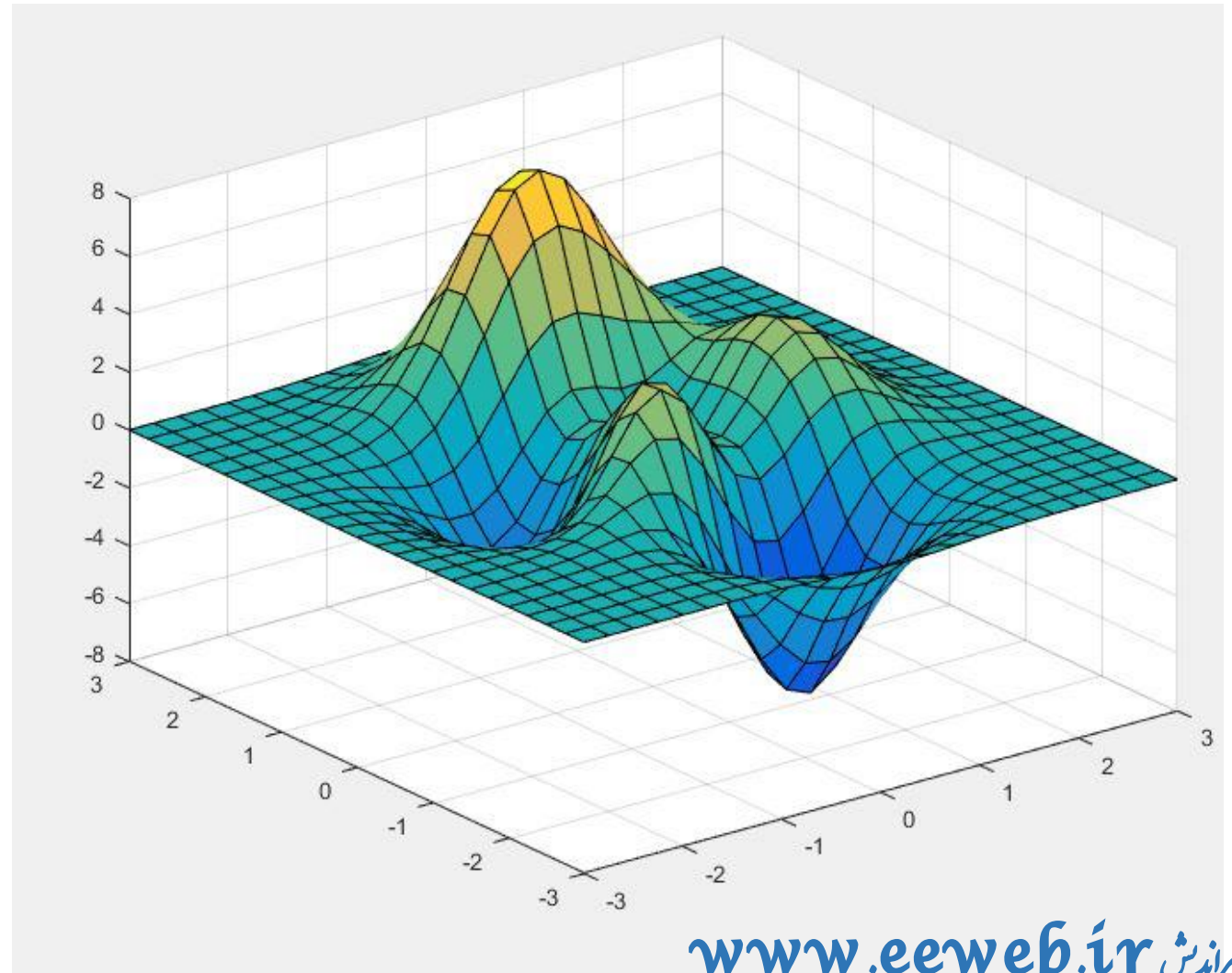
surf (x , y , z)

```
x=[1 -2 3; 5 9 7];  
y=[0 -2 3; -7 3 -9];  
z=sin(x)+cos(y);  
surf(z)
```



دستورات ترسیمات سه بعدی

```
[X,Y,Z] = peaks(25);  
surf(X , Y , Z);
```

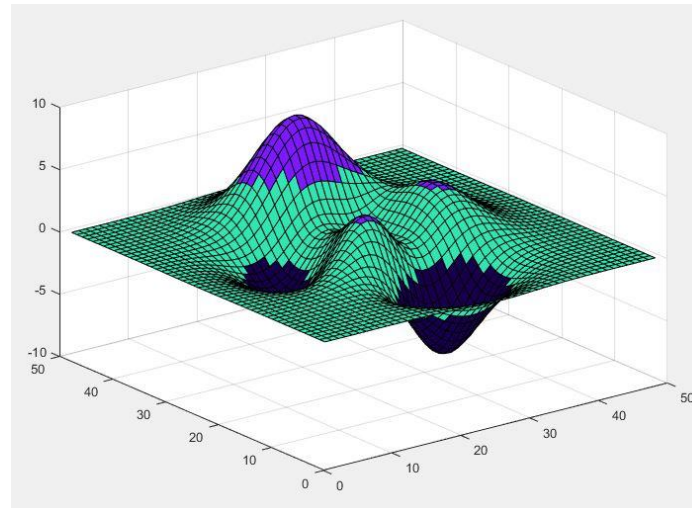


دستورات ترسیمات سه بعدی

colormap

این دستور برای تغییر رنگ پیش فرض یک شکل در متلب، به کار می رود .

colormap (X)



```
x = [0.1, 0, 0.3,  
      0.2, 0.9, 0.7,  
      0.5, 0.1, 1.0];  
surf(peaks)  
colormap(x)
```

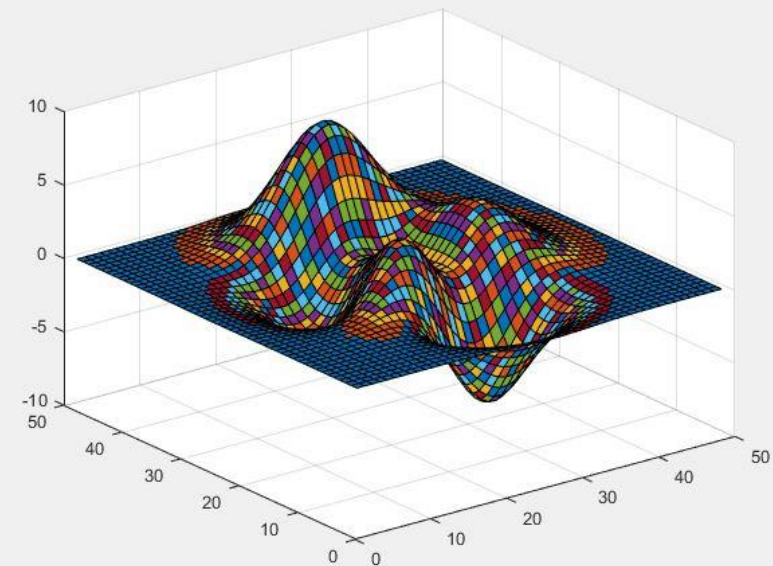
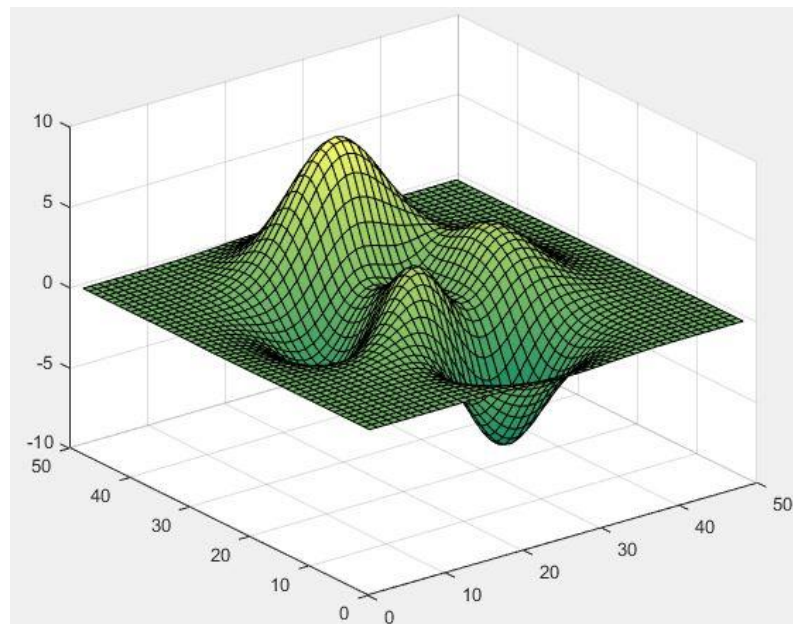
Built-In Colormap Functions



surf(peaks)
colormap summer

surf(peaks)
colormap lines

برخی مفاهیم رسم نمودار :

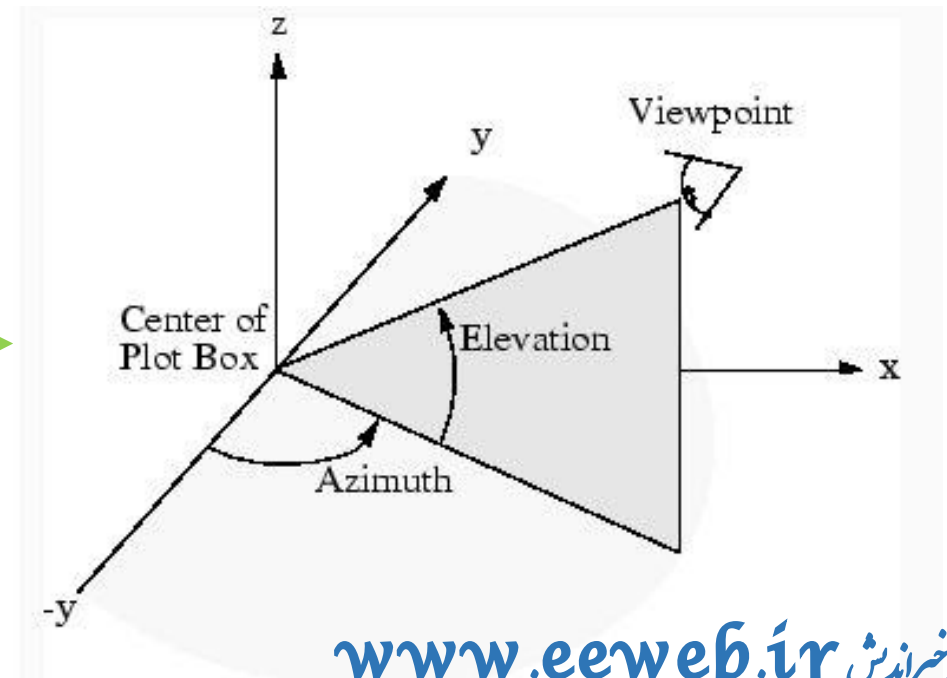


دستورات ترسیمات سه بعدی

این دستور جهت کنترل نمای یک نمودار سه بعدی مورد استفاده قرار می گیرد ، این کنترل از طریق زاویه el در بالای صفحه XY و چرخش محورهای مختصات با زاویه az در جهت ساعتگرد است .
مقادیر پیش فرض $el=30'$ و $az=37^{\circ}0.5'$

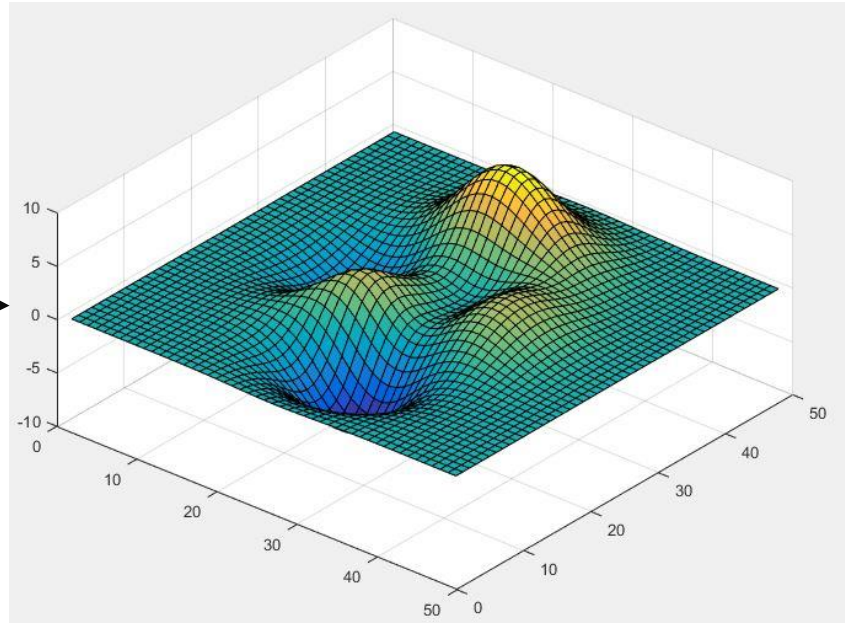
view

view (az , el)



برخی مفاهیم رسم نمودار :

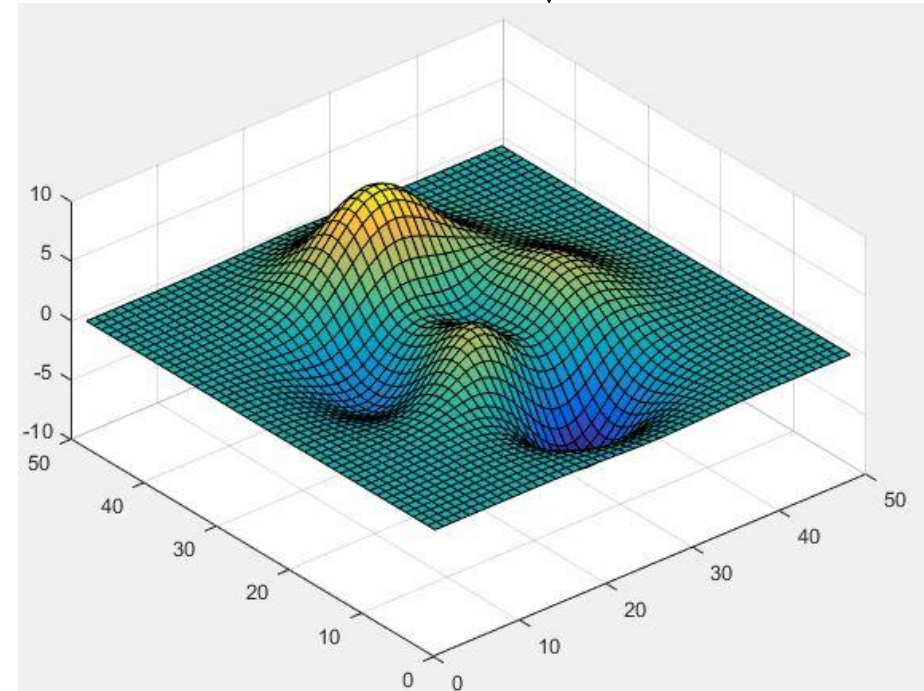
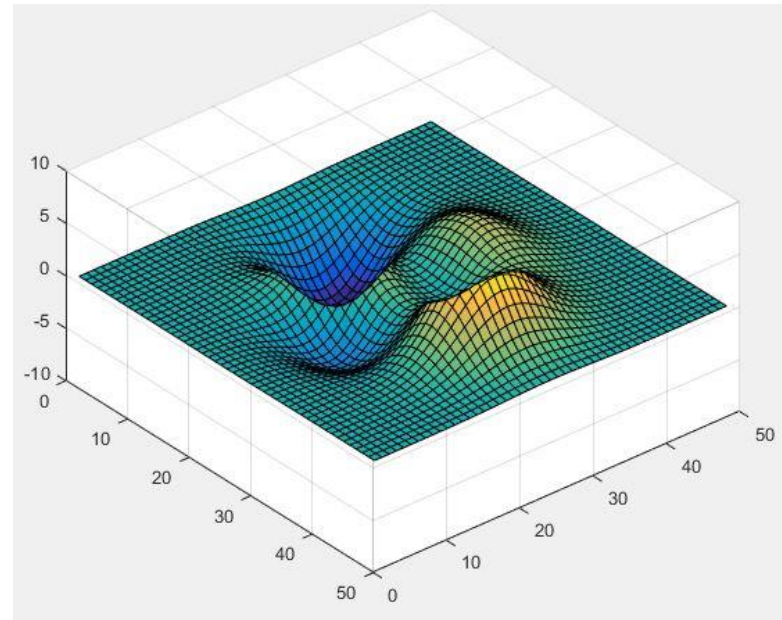
```
surf(peaks);  
view(40,50)
```



```
surf(peaks);  
view(-40,50)
```



```
surf(peaks);  
view(40, -50)
```



دستورات ترسیمات سه بعدی

با استفاده از این دستور می توان مقیاس یک نمودار را تعیین و یا تغییر داد. اگر مختصات محورها محدود به دو بعد یا سه بعد بوده و شامل بردار سطری :

$r=[Xmin, Xmax, Ymin, Ymax, Zmin, Zmax]$

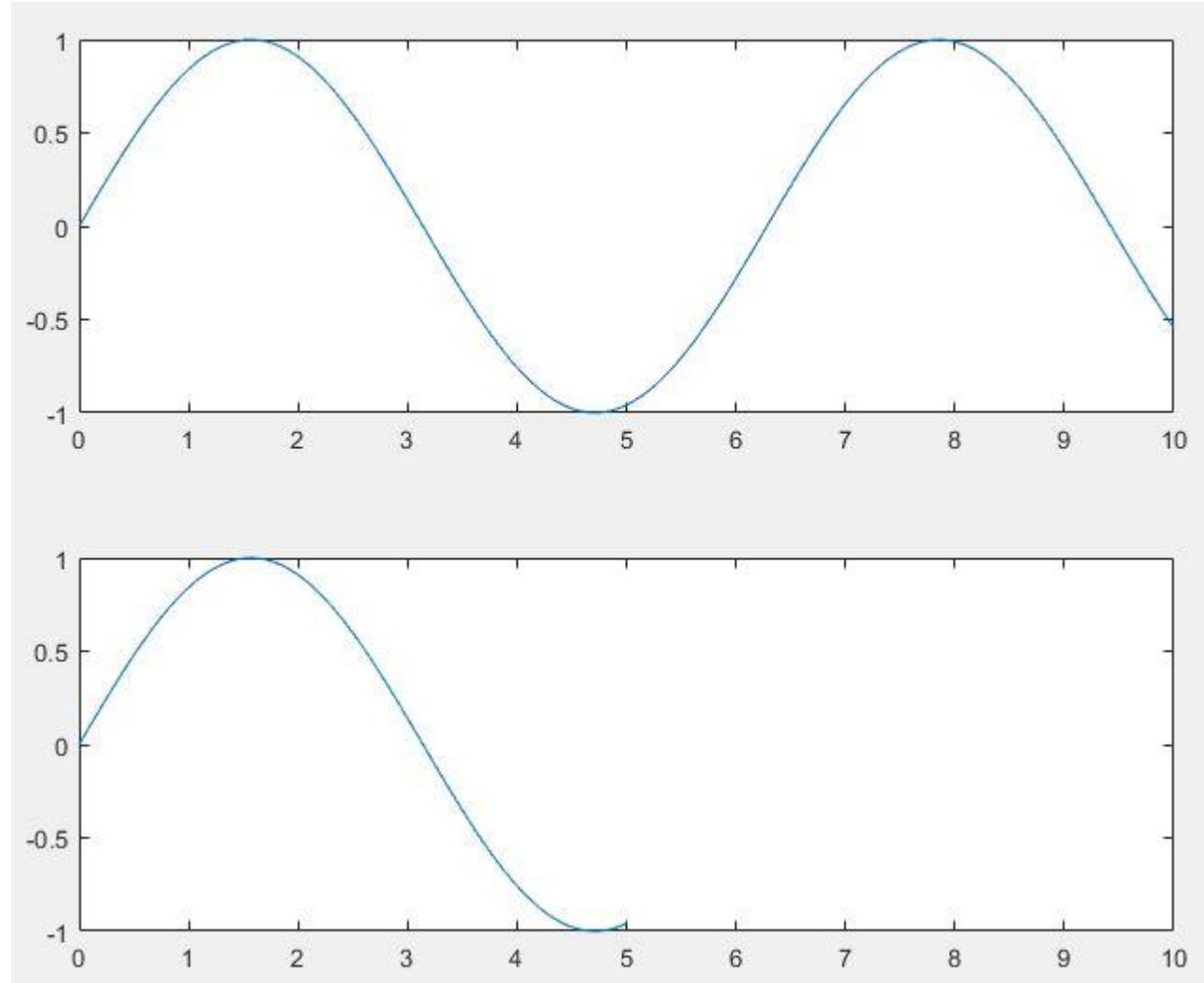
باشد، این دستور جهت تغییر مختصات محورها بر اساس بردار سطری r بکار می رود .

axis

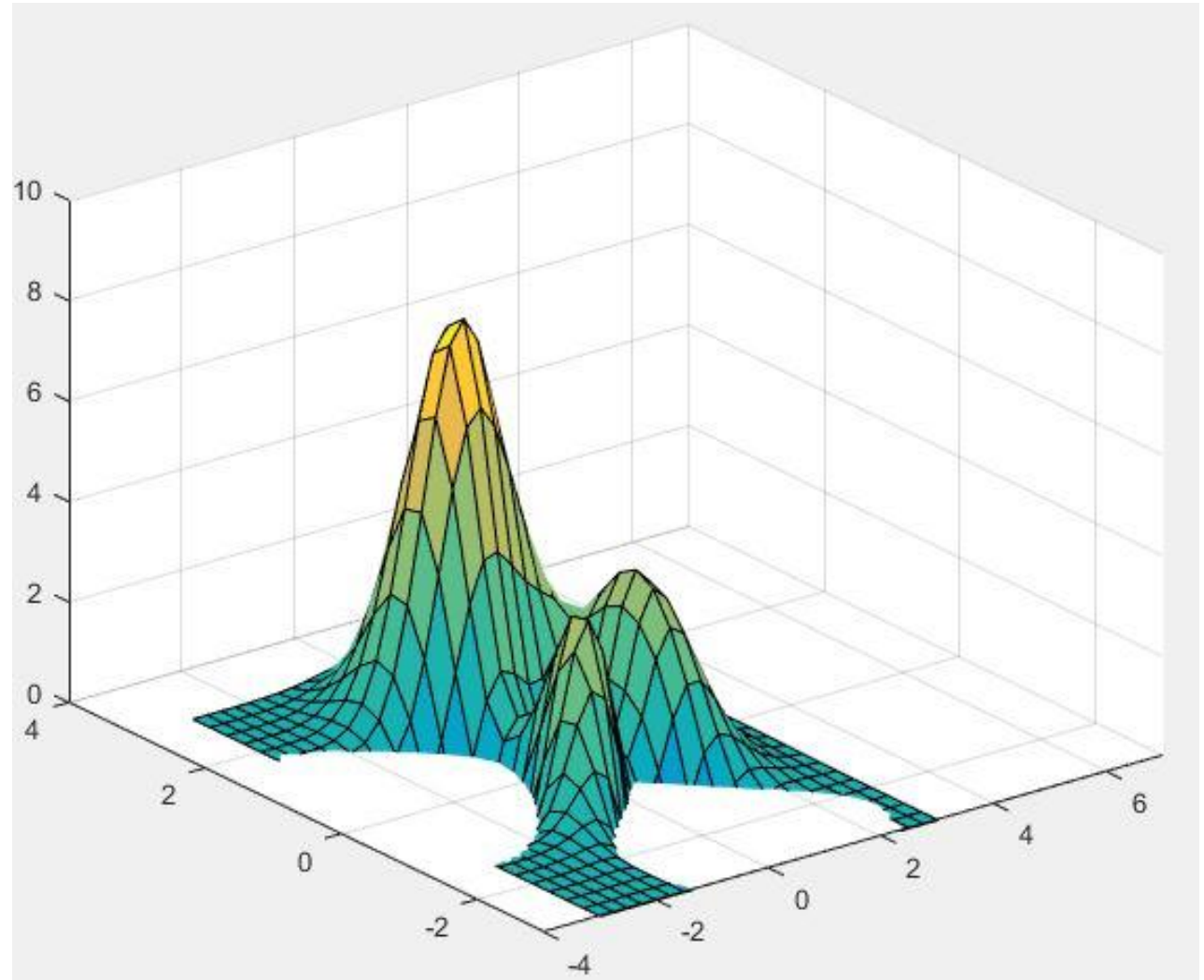
axis(r)

$r=[0 \ 1 \ 0 \ 2 \ 0 \ 5]$

```
x1 = linspace(0,10,100);  
y1 = sin(x1);  
ax1 = subplot(2,1,1);  
plot(ax1,x1,y1)  
x2 = linspace(0,5,100);  
y2 = sin(x2);  
ax2 = subplot(2,1,2);  
plot(ax2,x2,y2)  
axis([ax1 ax2],[0 10 -1 1])
```



```
[X,Y,Z] = peaks(25);  
surf(X , Y , Z);  
r=[-4 7 -3 4 0 10];  
axis(r)
```



`axis('equal')`

این دستور باعث می شود تا مقیاس x, y مشابه باشند.

`axis('square')`

تبدیل شکل محورها به صورت مربعی

`axis('auto')`

برگرداندن تنظیمات مقیاس محورها به تنظیمات پیش فرض

`axis('normal')`

تبدیل مقیاس محورها به بزرگترین اندازه

دستورات ترسیمات سه بعدی

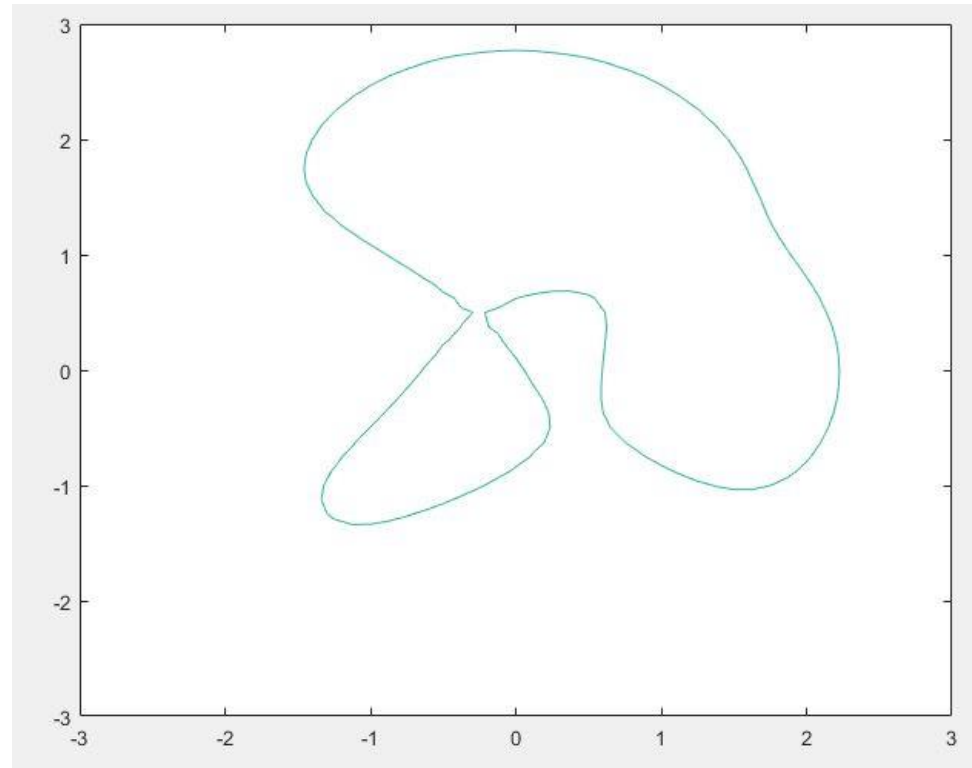
contour

(حد فاصل)
با استفاده از این دستور می توان خطوط هم فشار در
ماتریس Z را رسم کرد . همچنین با مشخص کردن n می
توان سطح هم فشاری را به دلخواه تعیین نمود .

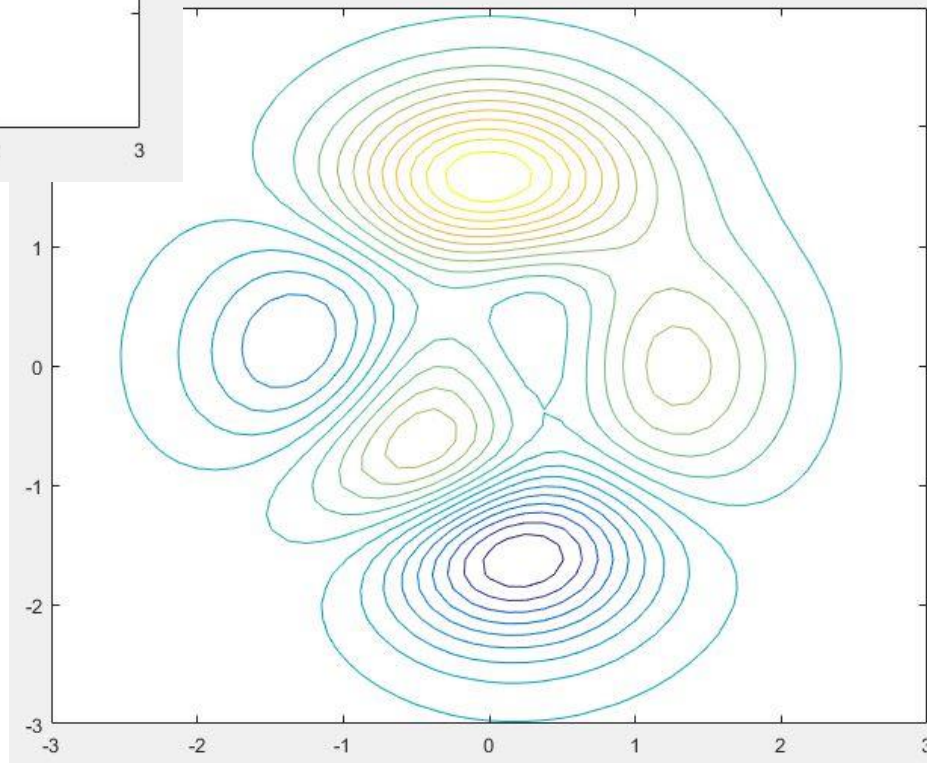
contour(z)
Contour(z,n)

```
X=[ ... ];  
Y=[ ... ];  
Z= ....X ....Y....;  
Contour(z,n)
```

```
[X,Y,Z] = peaks;  
contour(X,Y,Z,1)
```



```
[X,Y,Z] = peaks;  
contour(X,Y,Z,20)
```



دستورات ترسیمات سه بعدی

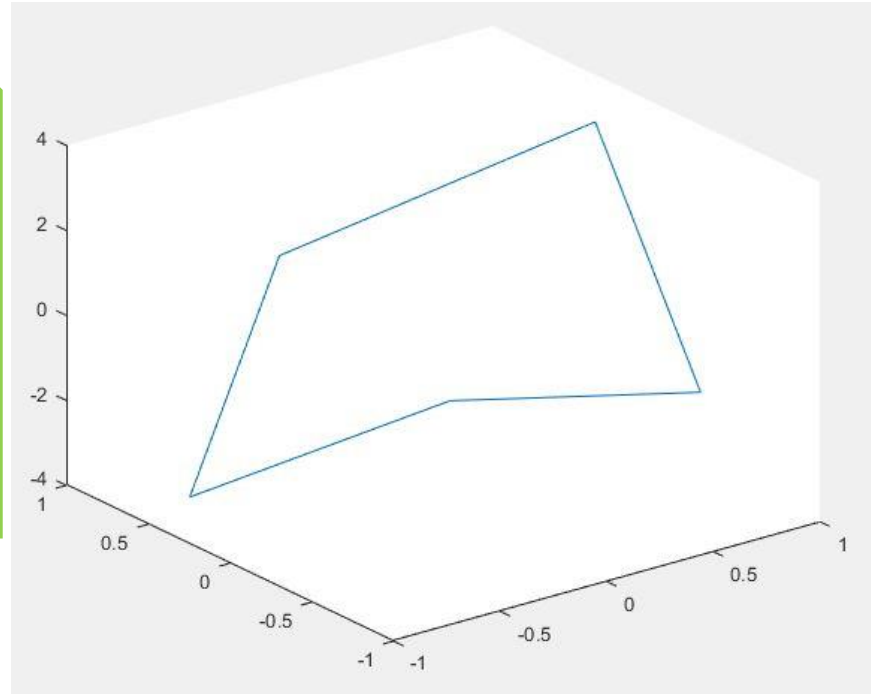
plot3

این دستور مشابه دستور plot بوده که اگر x, y, z بردارهایی با ابعاد یکسان باشند، این دستور خطوطی را با اتصال نقاطی به یکدیگر در سه بعد رسم می کند. آرگومان چهارم می تواند برای مشخص کردن رنگ و علامت ها مورد استفاده قرار بگیرد.

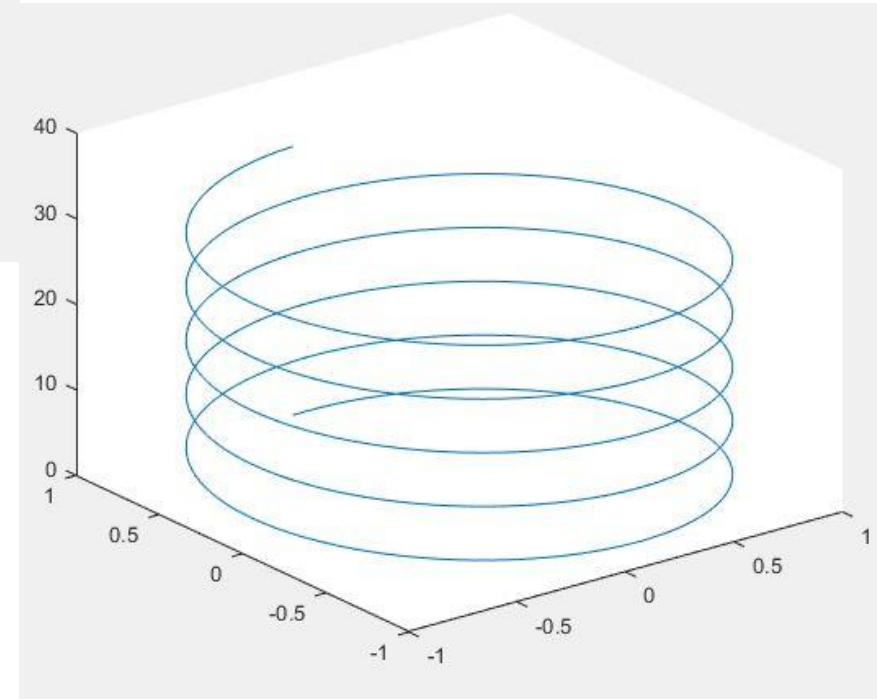
`plot3(x , y , z)`
`plot3(x , y , z , ...)`

```
X=[ ... ];  
Y=[ ... ];  
Z= [ ... ];  
Plot3(x,y,z,'-r'
```

```
t = 0:0.4:10;  
x = sin(pi*t);  
y = cos(pi*t);  
z = tan(pi*t);  
plot3(x,y,z)
```

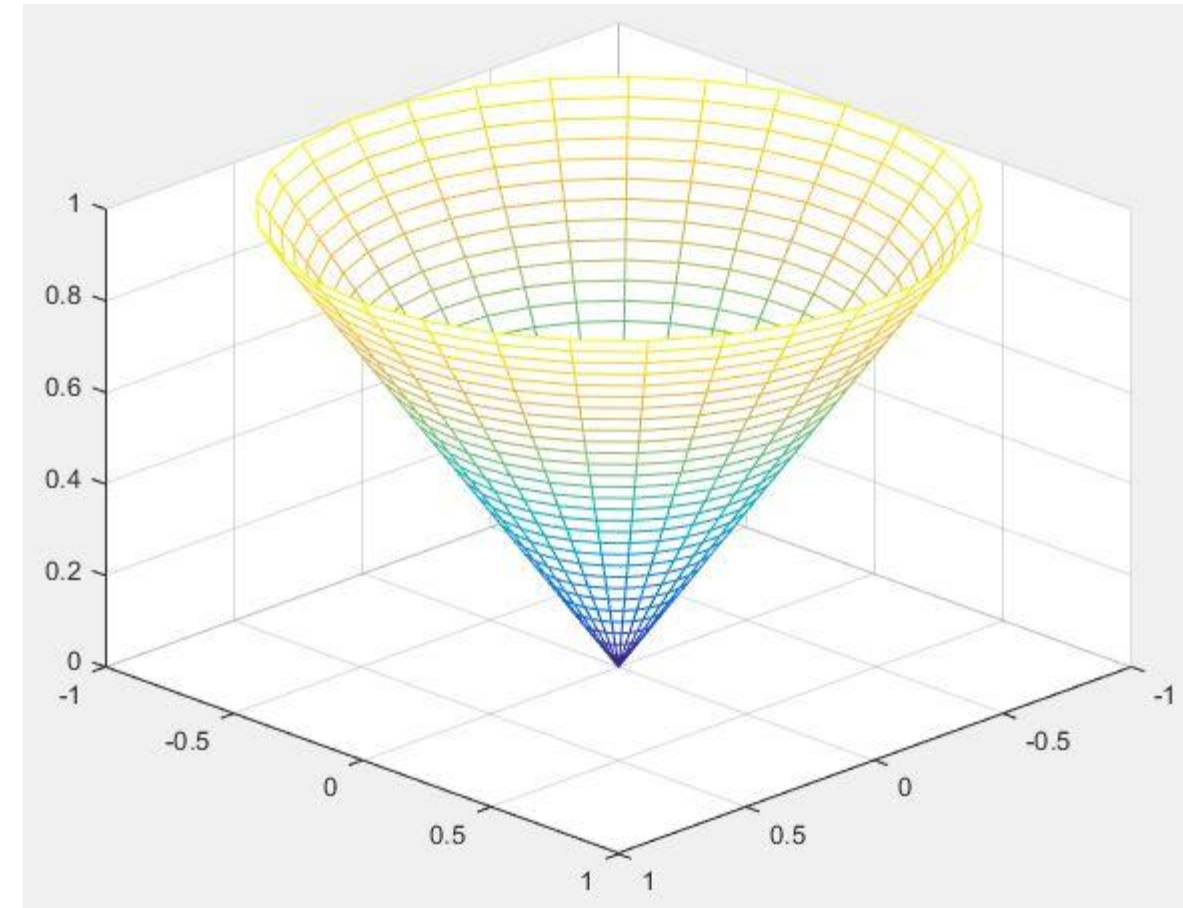


```
t = 0:pi/50:10*pi;  
x = sin(t);  
y = cos(t);  
plot3(x,y,t)
```



برخی ترسیمات سه بعدی

```
r=linspace(0,1,30);  
theta=linspace(0,2*pi,30);  
[r,theta]=meshgrid(r,theta);  
x=r.*cos(theta);  
y=r.*sin(theta);  
z=r;  
mesh(x,y,z)  
view(135,30)
```



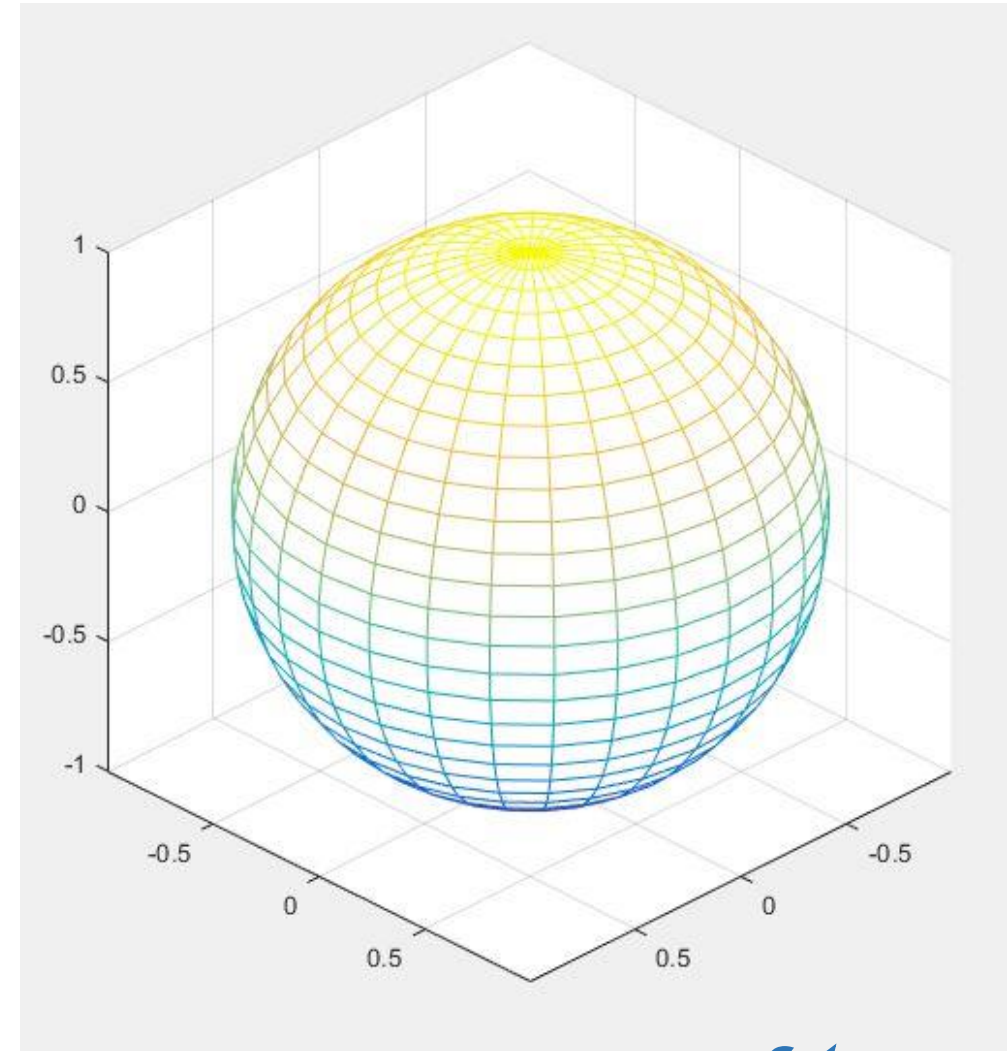
$$x = r \cos \theta, y = r \sin \theta, z = r$$

$$0 \leq r \leq 1 \quad 0 \leq \theta \leq 2\pi$$

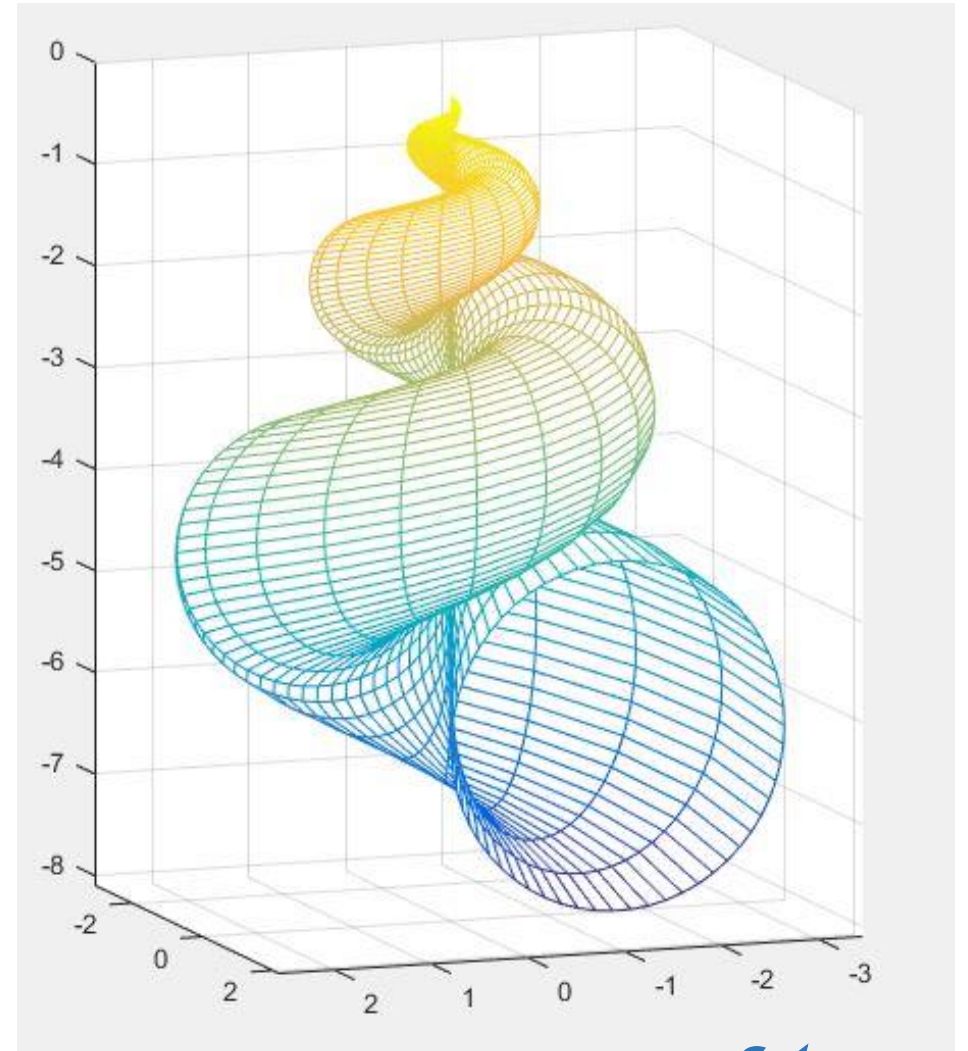
```
phi=linspace(0,pi,30);  
theta=linspace(0,2*pi,30);  
[phi,theta]=meshgrid(phi,theta);  
x=sin(phi) .*cos(theta);  
y=sin(phi) .*sin(theta);  
z=cos(phi);  
mesh(x,y,z)  
view(135,30)  
axis equal
```

$$x = \sin\phi \cos\theta, y = \sin\phi \sin\theta, z = \cos\phi$$

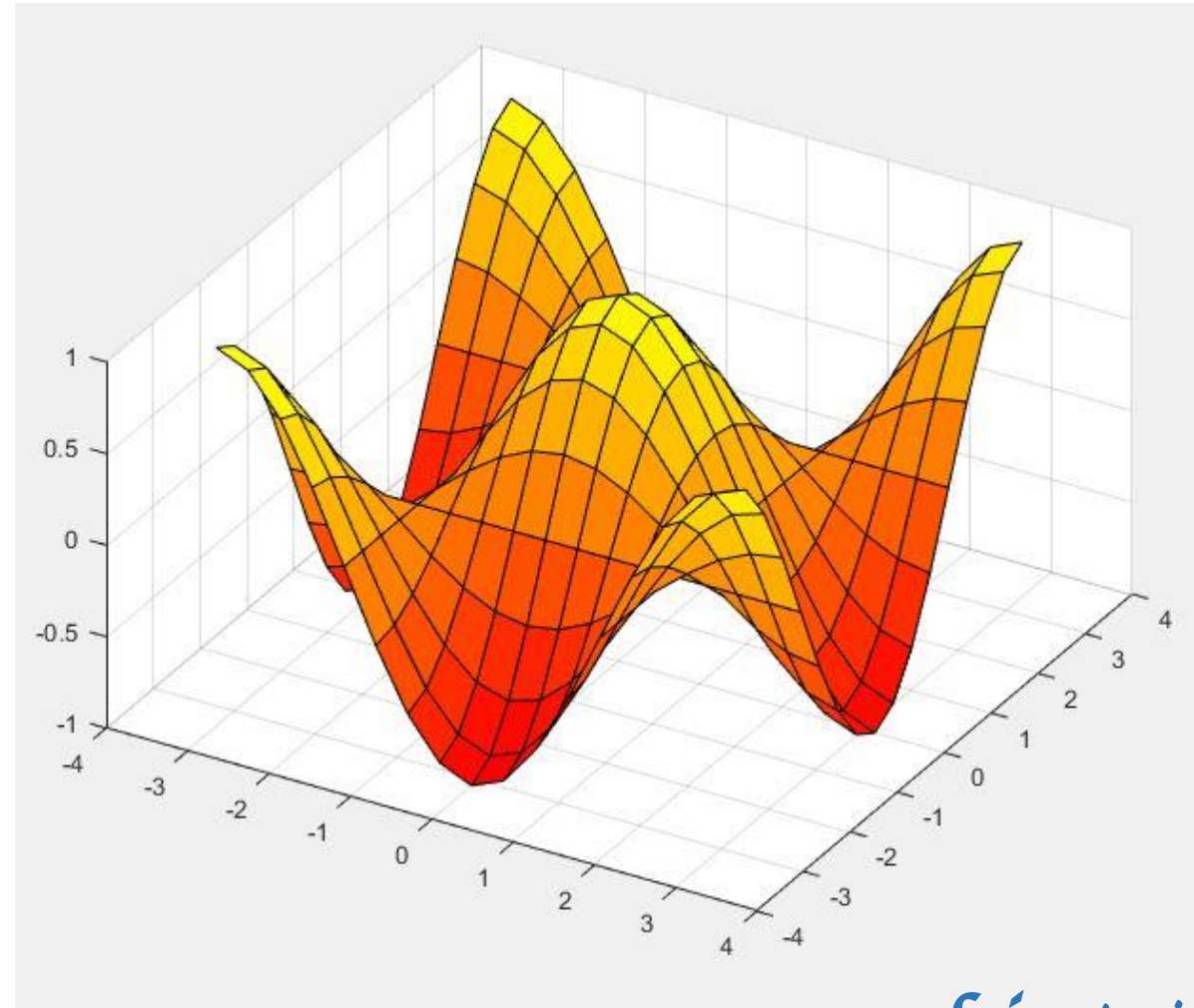
$$0 \leq \phi \leq 1 \quad 0 \leq \theta \leq 2\pi$$



```
u=linspace(0,6*pi,60);  
v=linspace(0,2*pi,60);  
[u,v]=meshgrid(u,v);  
x=2*(1-exp(u/(6*pi)) ).*cos(u) .*cos(v/2).^2;  
y=2*(-1+exp(u/(6 *pi) ) ) .*sin(u) .*cos(v/2) .^2;  
z=1-exp(u/(3*pi) ) - sin(v)+exp(u/(6*pi)).*sin(v);  
mesh(x,y,z)  
view(160,10)  
axis equal
```



```
[x,y]=meshgrid(-pi:pi/8:pi,-pi:pi/8:pi);  
z=cos(x).*cos(y);  
surf(x,y,z)  
view(30,45)  
colormap autumn
```



```
[x,y] = meshgrid(-2:2:2, -2:2:2);  
z = x .* exp(-x.^2 - y.^2);  
surf(x,y,z)  
colormap winter
```

