



MATLAB

(*MATRIX*
LABORATORY)

PLOTTING (ÇİZİM YAPMA)

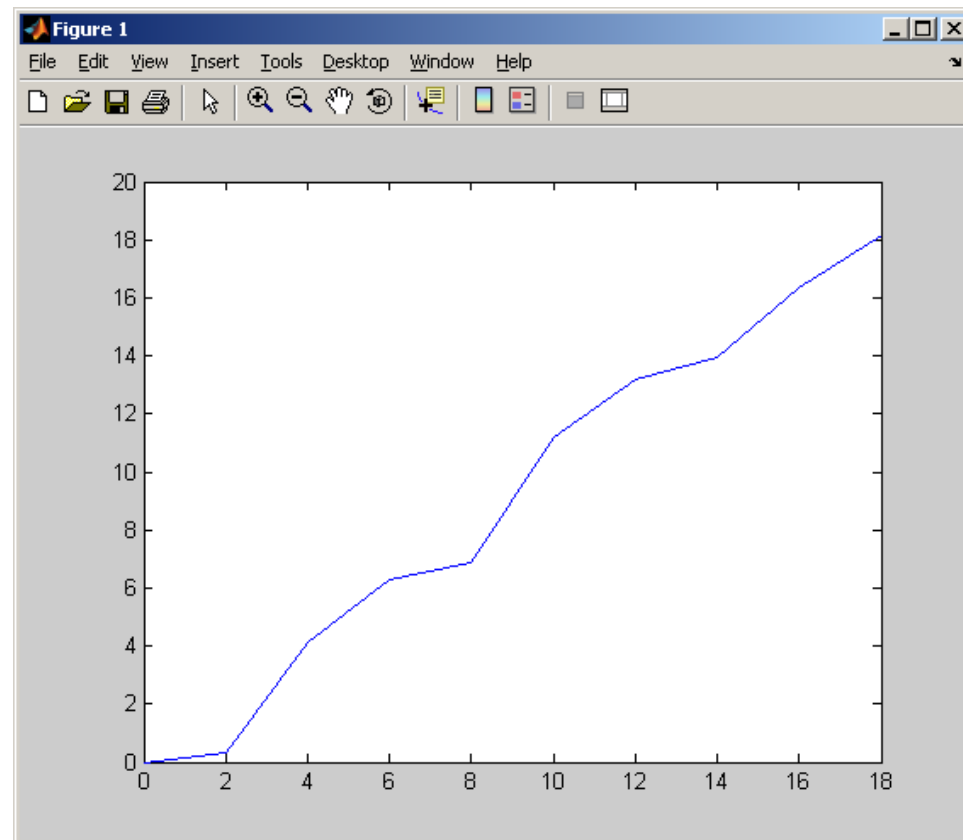
- Mühendisler tarafından kullanılan en yaygın çizim şekli x-y ekseninde yapılan çizimdir.
- Genellikle x bağımsız değişken, y de bağımlı değişken olur.

time, sec	Distance, Ft
0	0
2	0.33
4	4.13
6	6.29
8	6.85
10	11.19
12	13.19
14	13.96
16	16.33
18	18.17

Zaman bağımsız değişken,
uzaklık ise bağımlı değişken.



```
Command Window
File Edit Debug Desktop Window Help
>> x=[0:2:18];
>> y = [0, 0.33, 4.13, 6.29, 6.85, 11.19, ...
        13.19, 13.96, 16.33, 18.17];
>> plot(x,y)
```



- Başlık
- X eksen etiketi, varsa ölçü birimiyle
- Y eksen etiketi, varsa ölçü birimiyle

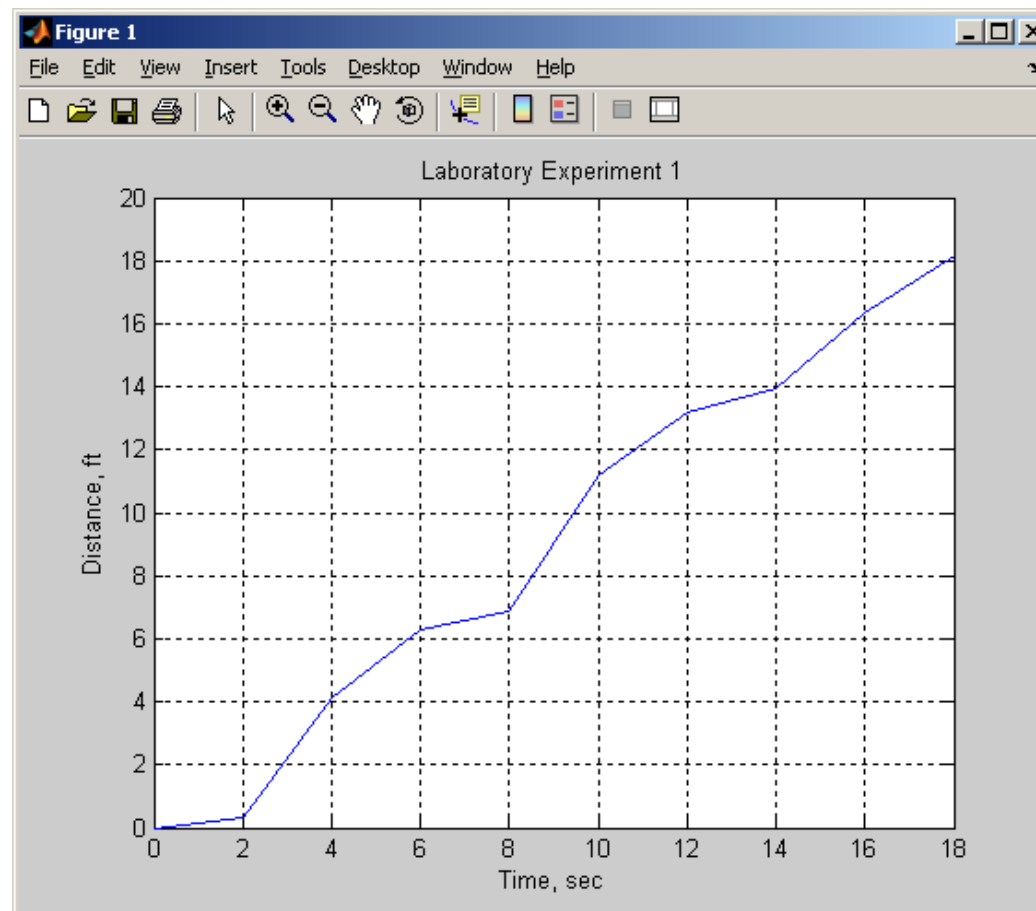
eklenmeli...

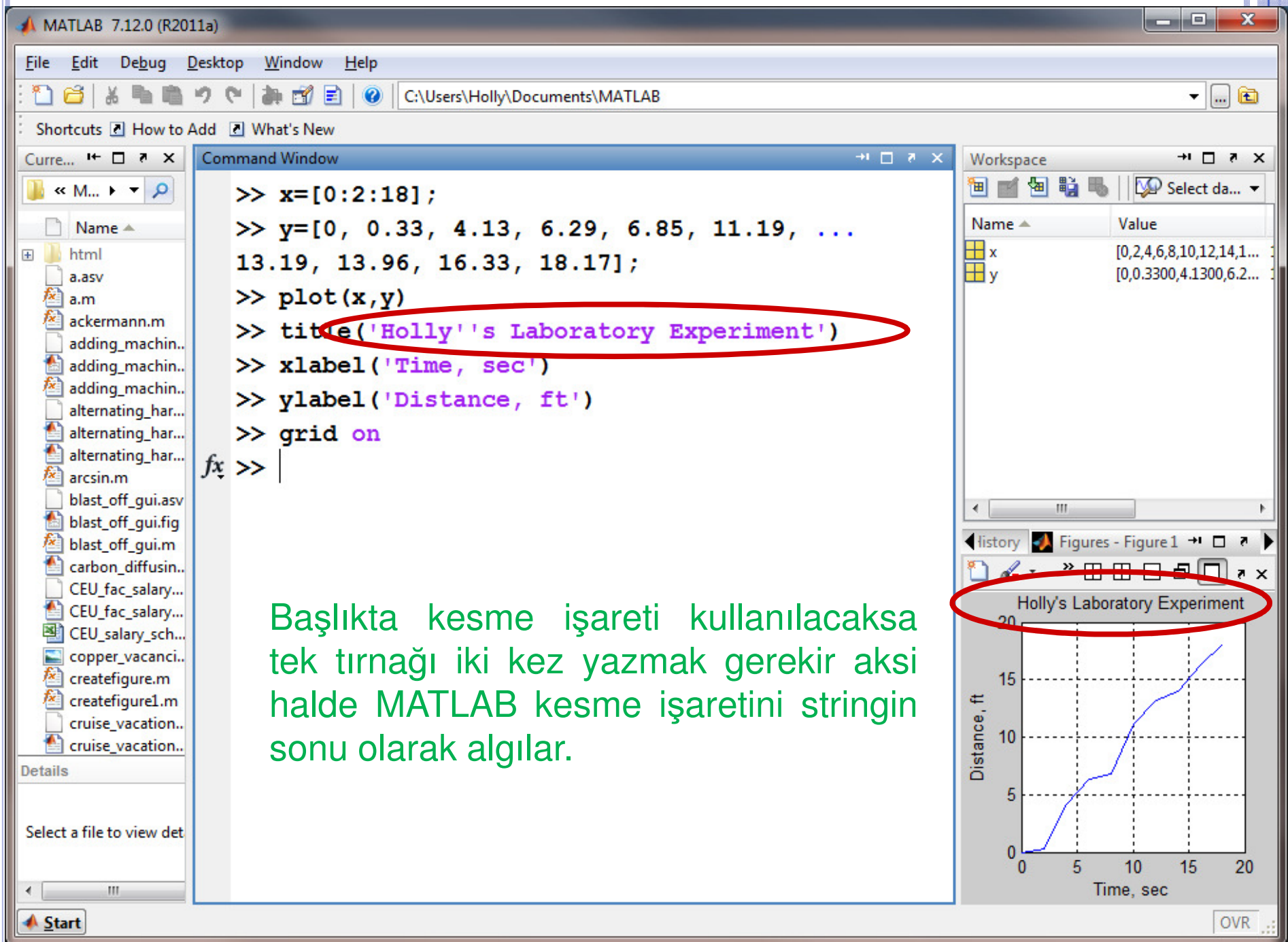
- Grid(ızgara) eklemek genellikle daha kullanışlı olur.



```
Command Window
File Edit Debug Desktop Window Help

>> title('Laboratory Experiment 1')
>> xlabel('Time, sec')
>> ylabel('Distance, ft')
>> grid on
```





- Her yeni plot komutuyla MATLAB varolan şekil penceresinin üzerine yazar.
- Yeni şekil penceresi açmak için figure fonksiyonunu kullanmak gerekir.



Şekil penceresinde birden fazla çizim yaptırmak:

- hold on

- O anki şekli dondurur ve üzerine yeni çizim eklenir.

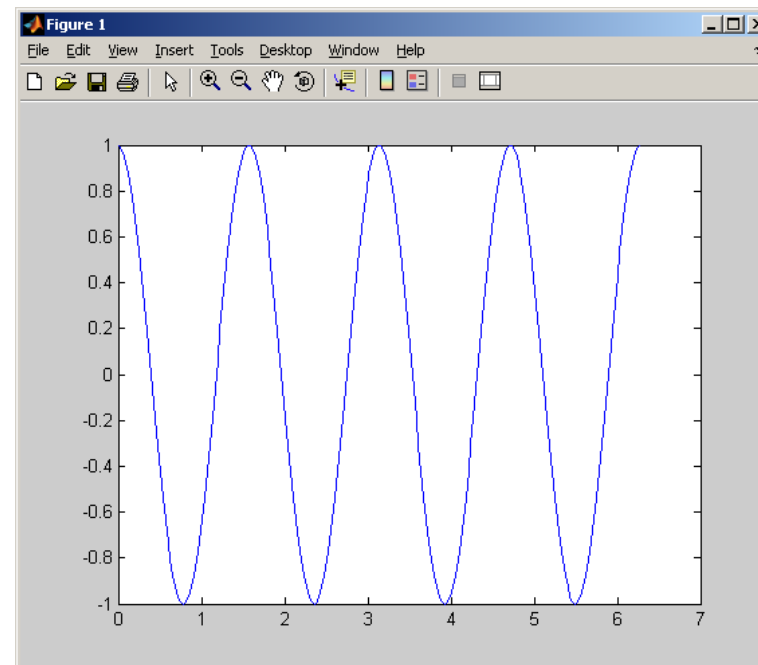
Bu komutu kullandığınızda eklenen yeni çizgi(line) default çizim rengi olan mavi renkle çizilir.




```
Command Window
File Edit Debug Desktop Window Help

>> x = 0:pi/100:2*pi;
>> y1 = cos(x*4);
>> plot(x,y1)
>>
```

İlk çizim mavi renkli olur



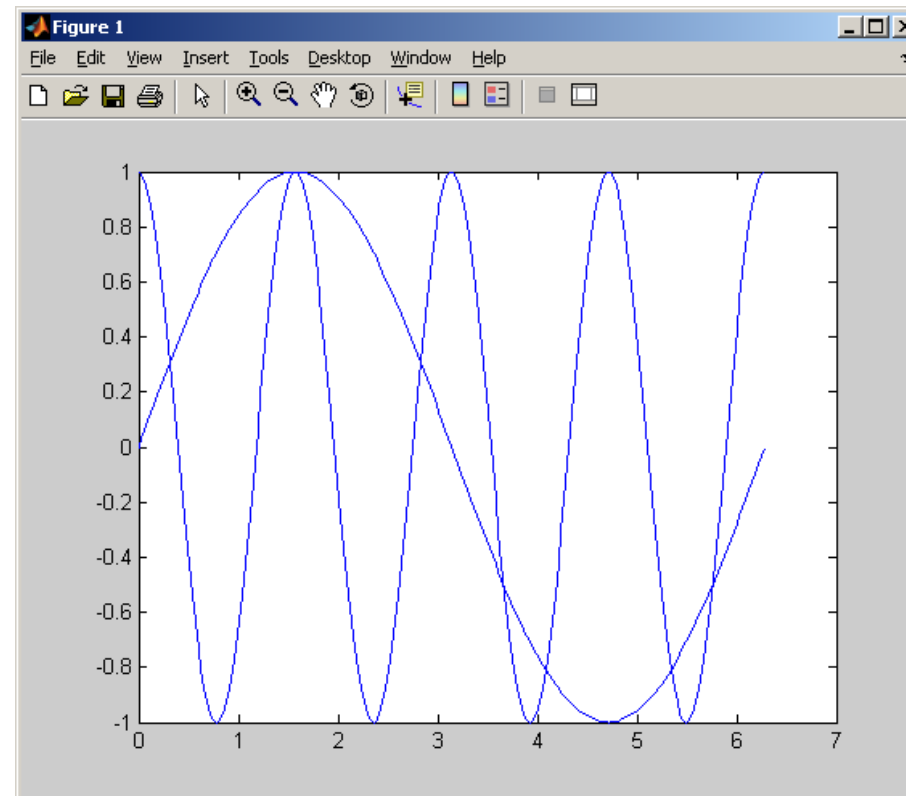
```
Command Window
File Edit Debug Desktop Window Help

>> x = 0:pi/100:2*pi;
>> y1 = cos(x*4);
>> plot(x,y1)
>> hold on
>> y2 = sin(x);
>> plot(x, y2)
```

hold on komutu çizimi
dondurur

İkinci eğri(line) de mavi
renkli olur.

Çizimi dondurmayı
sonlandırmak için hold
off komutu kullanılır.

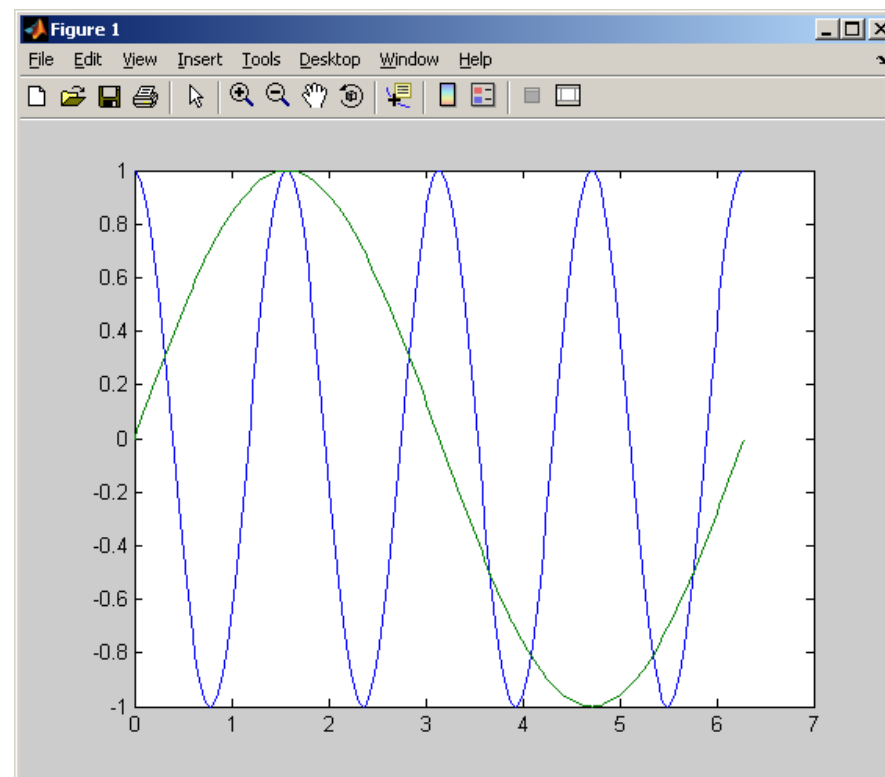


- Aynı zamanda tek grafik üzerinde çoklu çizimi, tek komutla da yapabilirsiniz.
- Bu yöntemi kullanmak her eğrinin farklı renkte çizimini sağlayacaktır.



```
Command Window
File Edit Debug Desktop Window Help

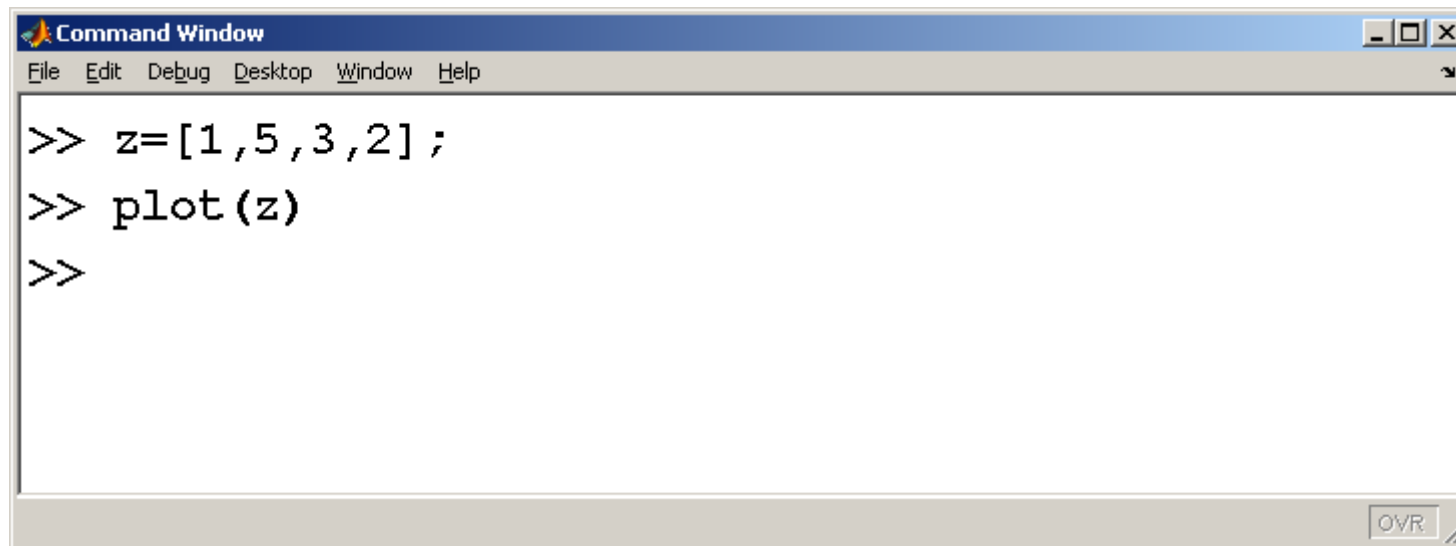
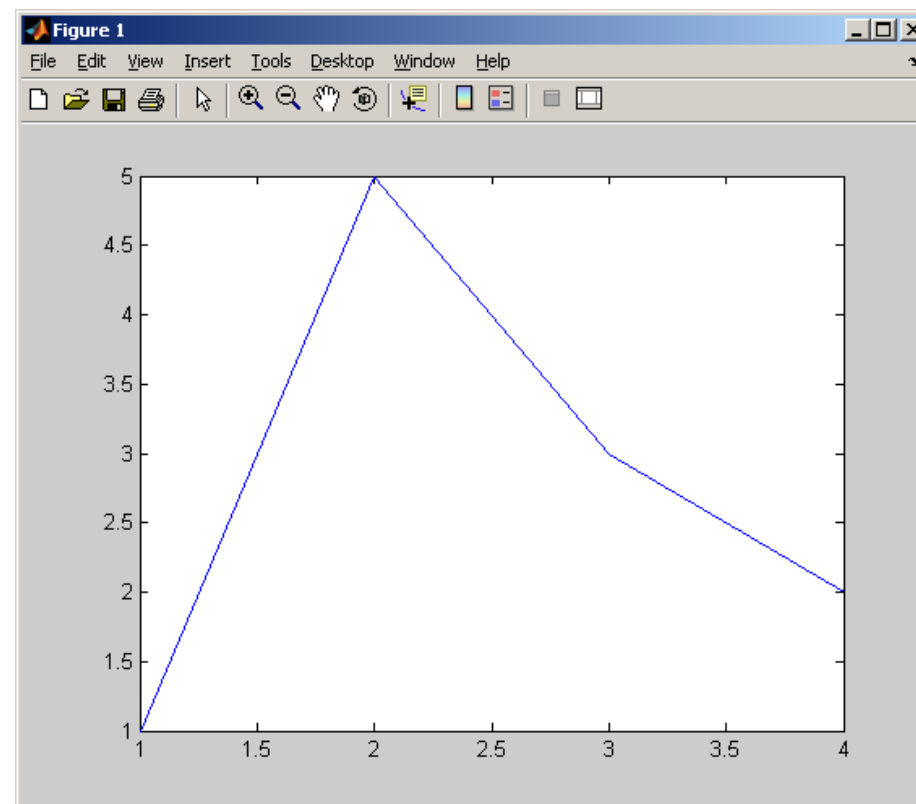
>> x = 0:pi/100:2*pi;
>> y1 = cos(x*4);
>> y2 = sin(x);
>> plot(x,y1,x,y2)
>>
```



- Plot komutu tek bir matrisle kullanılıyorsa, MATLAB matris değerlerini, index değerlerinin karşılığı olacak şekilde çizer.
- Genellikle bu tür veriler çubuk grafik üzerinde çizdirilir.
- x-y eksenini üzerinde çizildiğinde buna eğri - çizgi(line) grafiği denir.



```
Command Window
File Edit Debug Desktop Window Help
>> z=[1,5,3,2];
>> plot(z)
>>
```

The image shows a MATLAB Command Window. The title bar is "Command Window" with standard window controls. The menu bar includes File, Edit, Debug, Desktop, Window, and Help. The command prompt shows three lines: ">> z=[1,5,3,2];", ">> plot(z)", and ">>". The window has a status bar at the bottom right with the text "OVR".

- Aynı x değeriyle çoklu çizim yapmak için

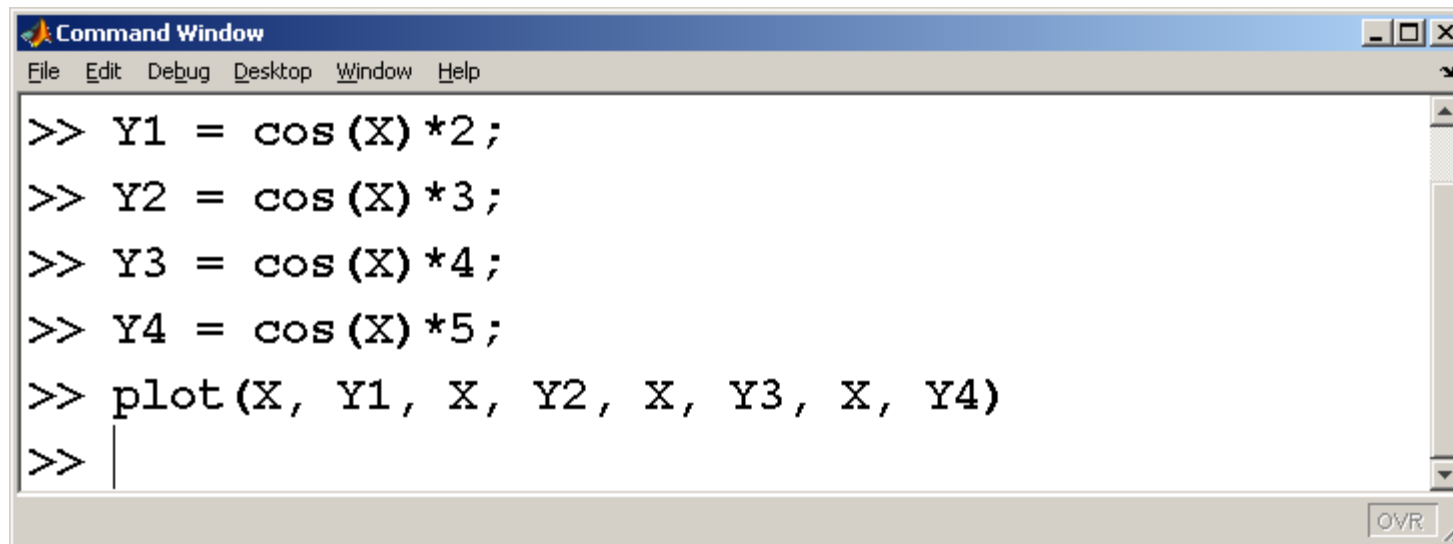
- `plot(x, y1, x, y2, x, y3, x, y4)`
- Ya da y değerleri bir matrise yazılır

`z = [y1, y2, y3, y4]`

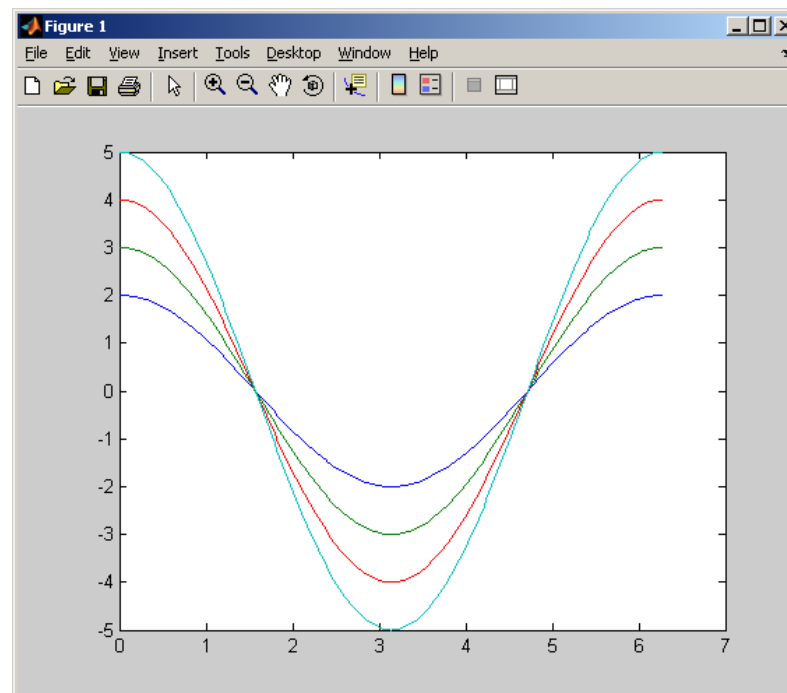
`plot (x,z)`



```
>> x = 0 : pi/100 : 2*pi;
```

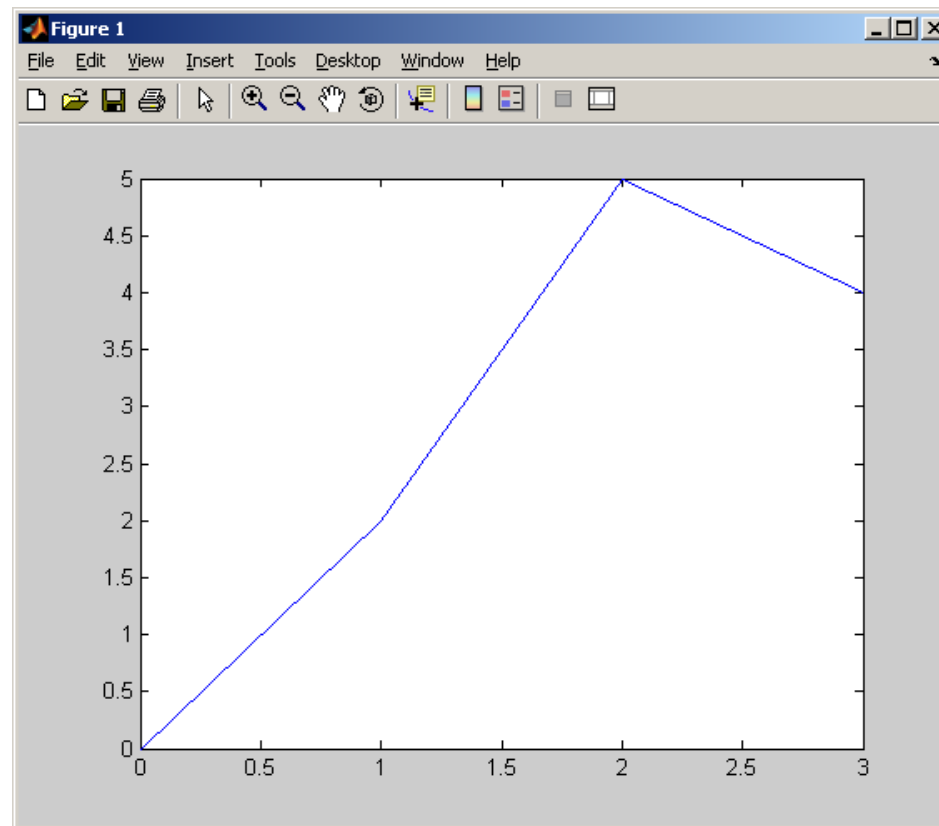


A screenshot of the MATLAB Command Window. The title bar reads "Command Window". The menu bar includes "File", "Edit", "Debug", "Desktop", "Window", and "Help". The command history shows the following commands: `>> Y1 = cos(X)*2;`, `>> Y2 = cos(X)*3;`, `>> Y3 = cos(X)*4;`, `>> Y4 = cos(X)*5;`, `>> plot(X, Y1, X, Y2, X, Y3, X, Y4)`, and `>>` followed by a cursor. A status bar at the bottom right shows "OVR".



- Plot komutunun girdisi kompleks sayılardan oluşan bir diziye, MATLAB gerçel kısmı x ekseninde, sanal kısmı ise y ekseninde çizer.

```
Command Window
File Edit Debug Desktop Window Help
>> A=[0+0i,1+2i, 2+5i, 3+4i];
>> plot(A)
```

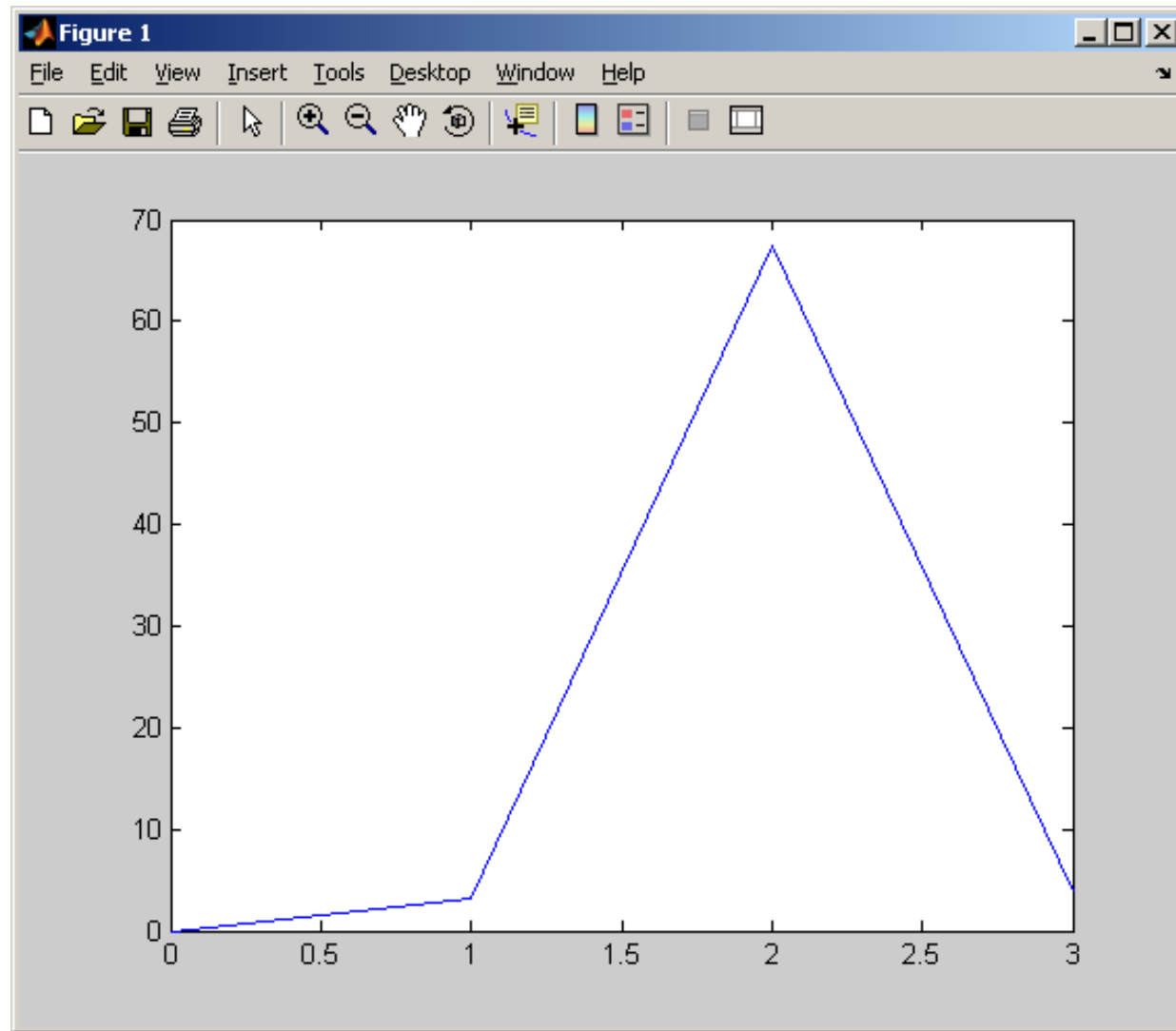


Kompleks sayılardan oluşan iki diziyi plot fonksiyonunda kullanmak istersek, sanal kısımlar görmezden geliriz.

```
Command Window
File Edit Debug Desktop Window Help

>> A=[0+0i,1+2i, 2+5i, 3+4i];
>> B=sin(A)
B =
Columns 1 through 2
0 3.1658 + 1.9596i
Columns 3 through 4
67.4789 -30.8794i 3.8537 -27.0168i
>> plot(A,B)
Warning: Imaginary parts of complex X and/or Y arguments ignored.
>>
```





Grafiğinizin çizgi şekillerini, rengini, işaretleme şeklini değiştirmeniz mümkün. `help plot` komutu yardımıyla mümkün olan şekillerin listesini görebilirsiniz.

Table 5. 2 Line, Mark and Color Options

Line Type	Indicator	Point Type	Indicator	Color	Indicator
solid	-	point	.	blue	b
dotted	:	circle	o	green	g
dash-dot	-.	x-mark	x	red	r
dashed	--	plus	+	cyan	c
		star	*	magenta	m
		square	s	yellow	y
		diamond	d	black	k
		triangle down	v		
		triangle up	^		
		triangle left	<		
		triangle right	>		
		pentagram	p		
		hexagram	h		



plot(x,y,':ok')

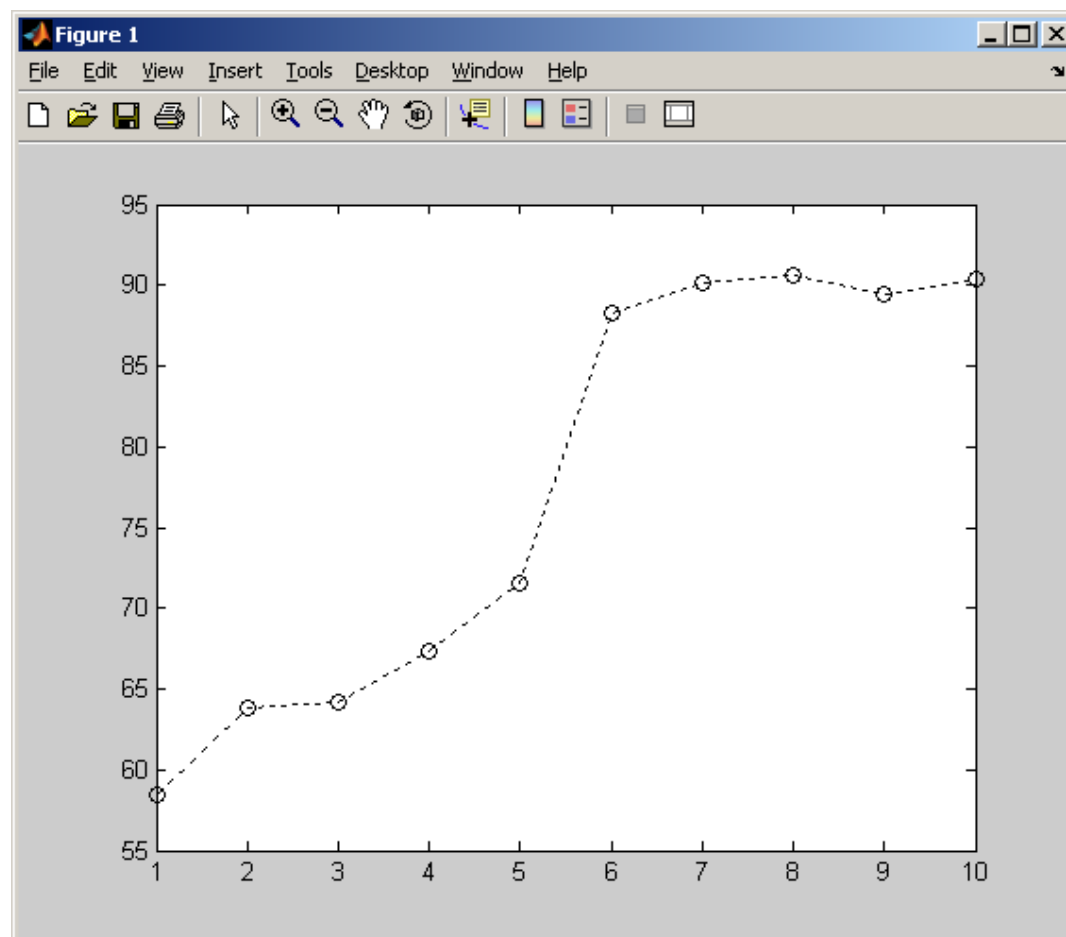
Bu komutta:

- :** → noktalı çizgi kullanılacağı anlamına gelir
- o** → Kesişen her noktayı işaretlemek için daire kullanır
- k** → Grafiğin siyah renkle çizileceği anlamına gelir



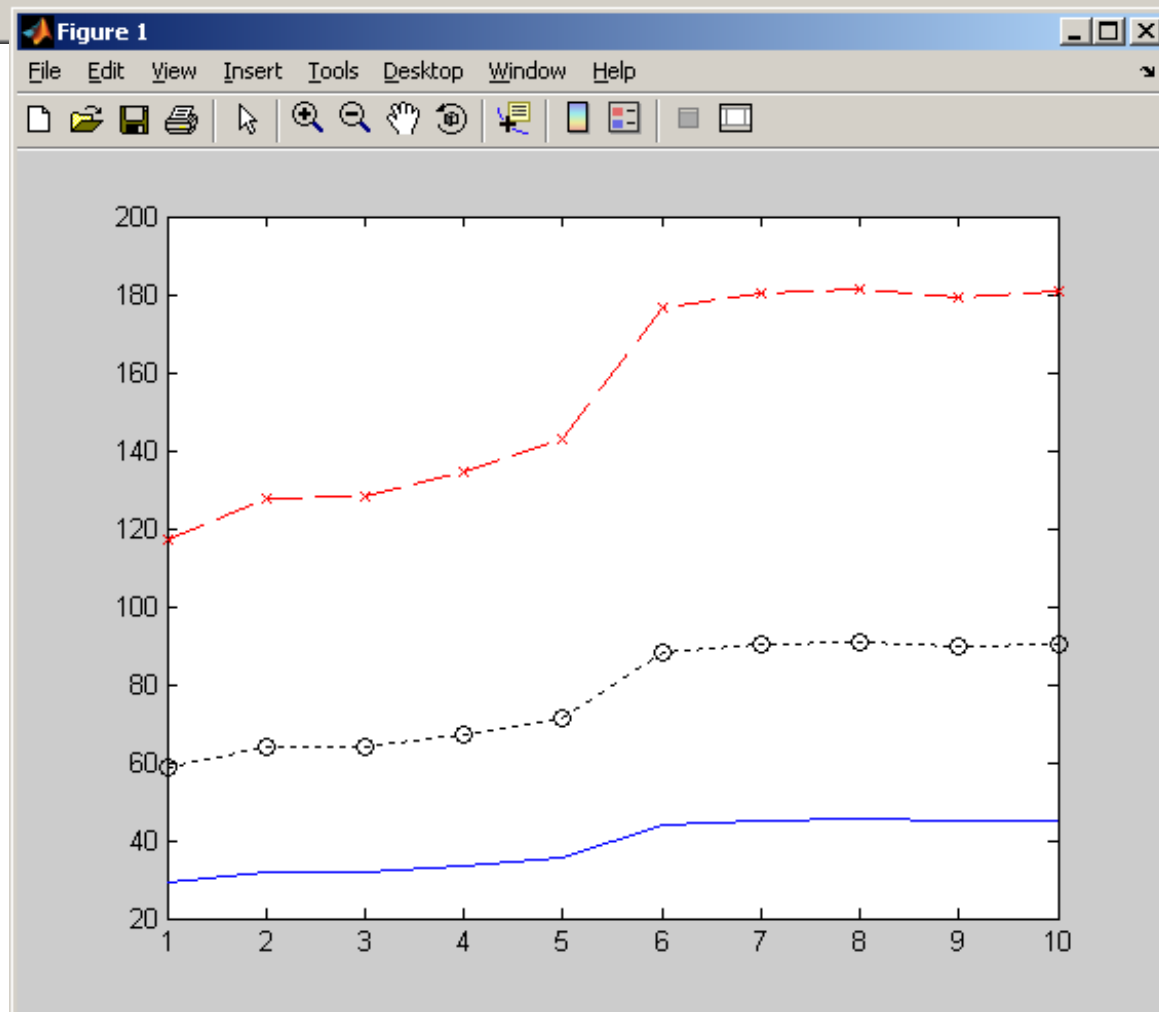
```
Command Window
File Edit Debug Desktop Window Help

>> x = [1:10];
>> y = [ 58.5, 63.8, 64.2, 67.3, 71.5, 88.3, ...
        90.1, 90.6, 89.5, 90.4];
>> plot(x,y, ':ok')
>>
```



```
Command Window
File Edit Debug Desktop Window Help

>> x = [1:10];
>> y = [ 58.5, 63.8, 64.2, 67.3, 71.5, 88.3,...
        90.1, 90.6, 89.5, 90.4];
>> plot(x,y,':ok',x,y*2,'--xr',x,y/2,'-b')
>>
```



Eksen Ölçeklendirme

MATLAB grafiği tamamen doldurmak için her şekli otomatik olarak ölçeklendirir.

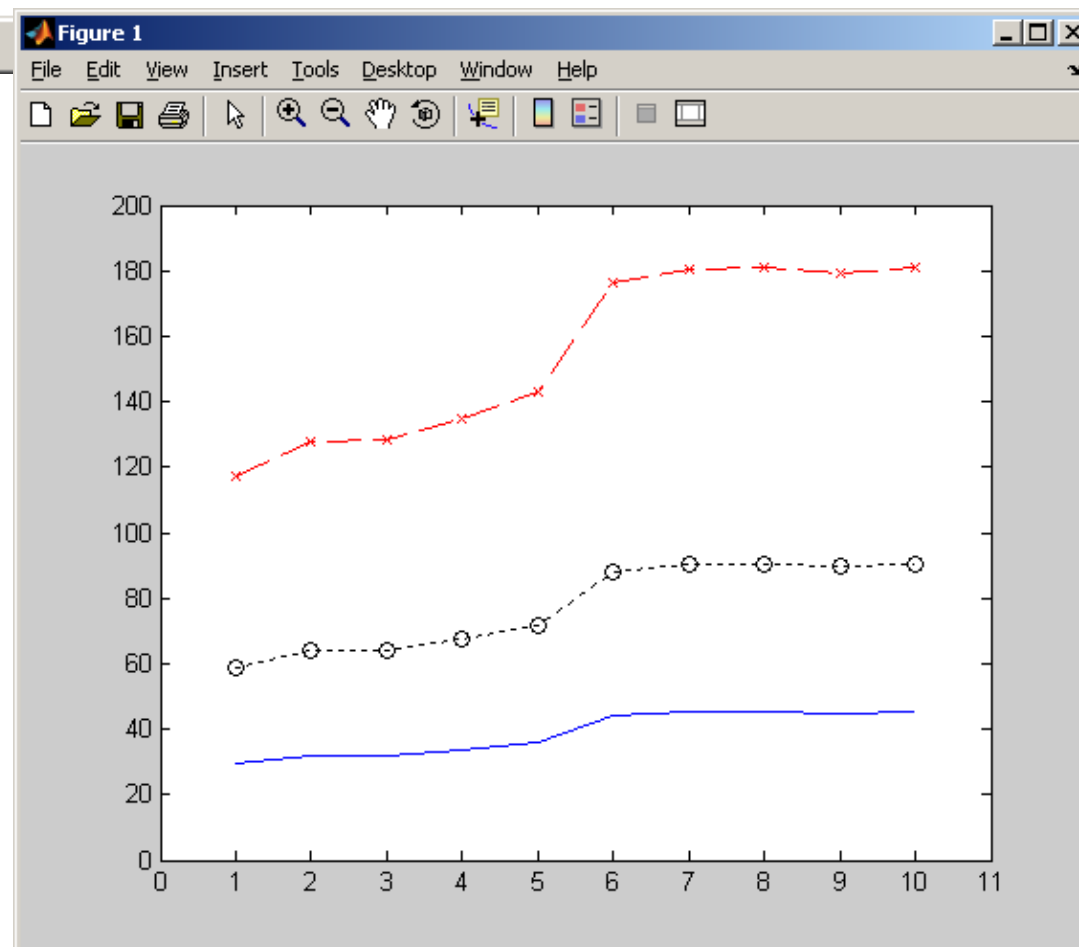
Farklı bir eksen belirlemek isterseniz axis komutunu kullanabilirsiniz.

axis([xmin,xmax,ymin,ymax])

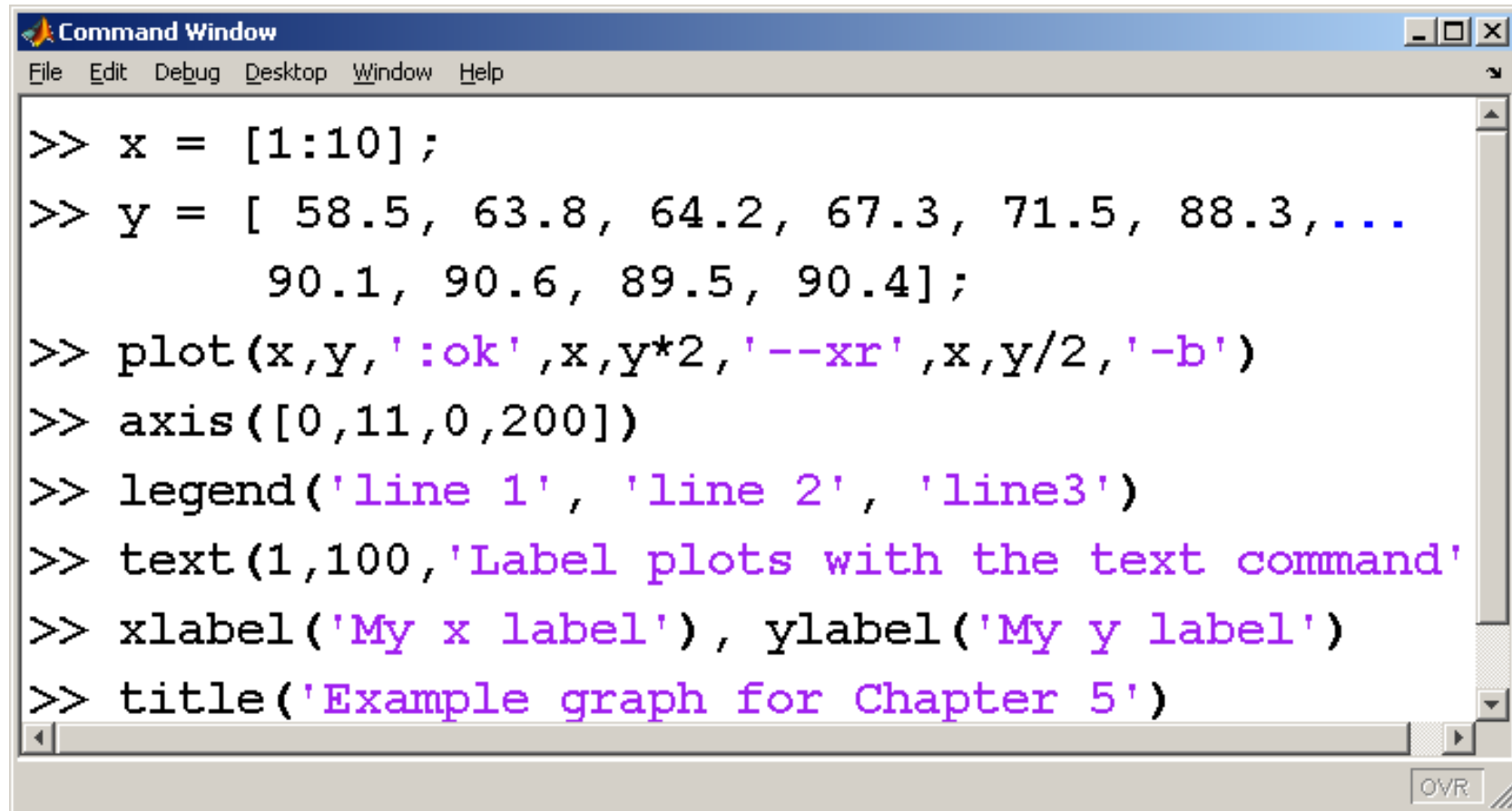



```
Command Window
File Edit Debug Desktop Window Help

>> x = [1:10];
>> y = [ 58.5, 63.8, 64.2, 67.3, 71.5, 88.3, ...
        90.1, 90.6, 89.5, 90.4];
>> plot(x,y,':ok',x,y*2,'--xr',x,y/2,'-b')
>> axis([0,11,0,200])
>>
```



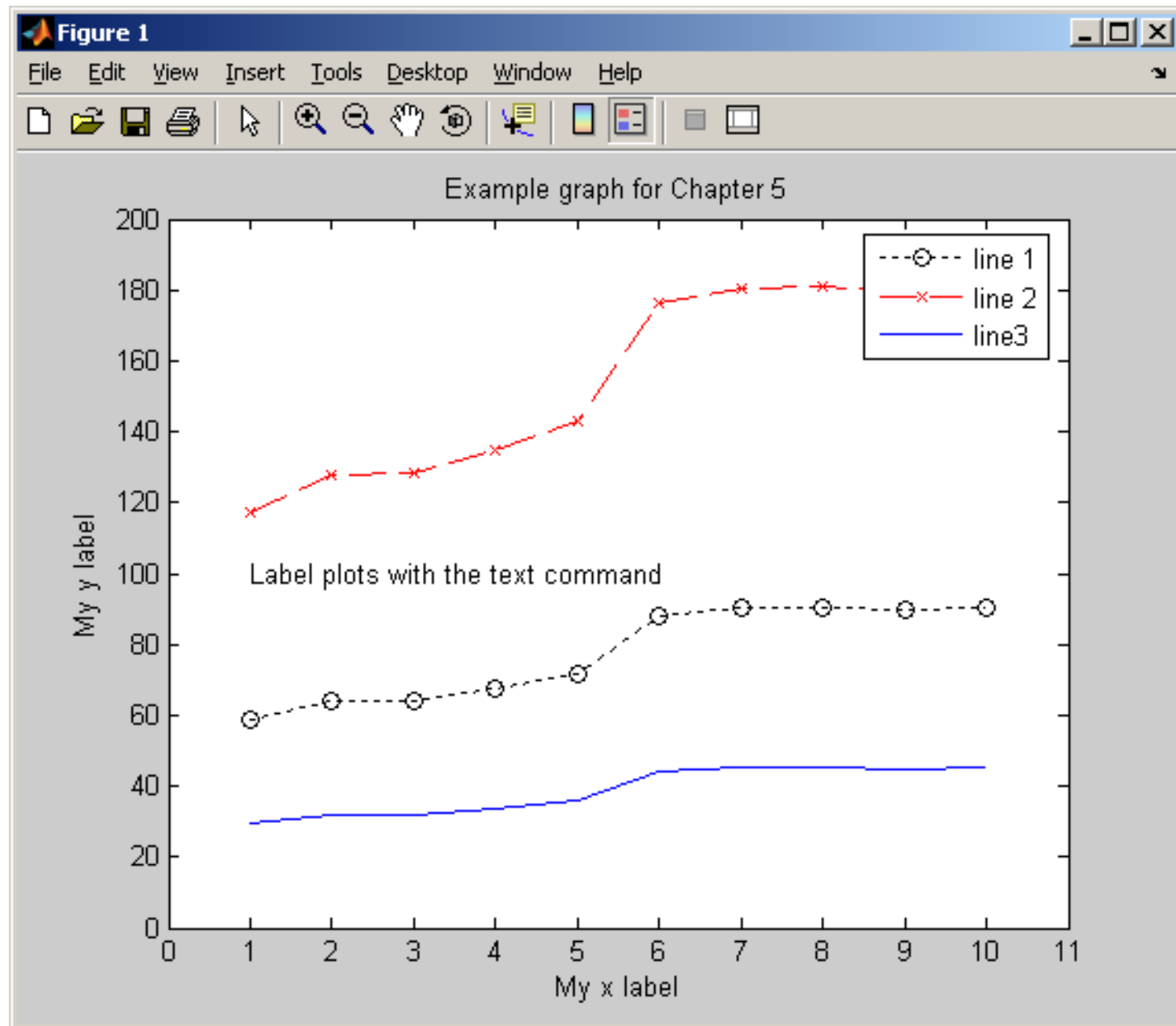
Aynı zamanda grafiğe açıklayıcı işaretler ve metin kutusu ekleyebilirsiniz.



```
Command Window
File Edit Debug Desktop Window Help

>> x = [1:10];
>> y = [ 58.5, 63.8, 64.2, 67.3, 71.5, 88.3, ...
        90.1, 90.6, 89.5, 90.4];
>> plot(x,y,':ok',x,y*2,'--xr',x,y/2,'-b')
>> axis([0,11,0,200])
>> legend('line 1', 'line 2', 'line3')
>> text(1,100,'Label plots with the text command')
>> xlabel('My x label'), ylabel('My y label')
>> title('Example graph for Chapter 5')
```

OVR



Etiketlerinizde Yunan harflerini kullanabilirsiniz. Bunun için harfin adından önce ters slash (\) işareti koymak gerekir.

Örneğin;

title (' \alpha \beta \gamma ')

komutuyla grafiğin başlığı:

$\alpha \beta \gamma$

şeklinde olur.



- Üst simge oluşturmak için şapka sembolü kullanılır:

title ('x^2')

komutu

x^2

sonucunu verir.

- Alt simge oluşturmak için altçizgi kullanılır:

title ('x_2')

komutu

x_2

sonucunu verir.

- Alt simge ya da üst simge bir karakterden fazla olacaksa süslü parantez kullanılır:

title('x^{-2}')



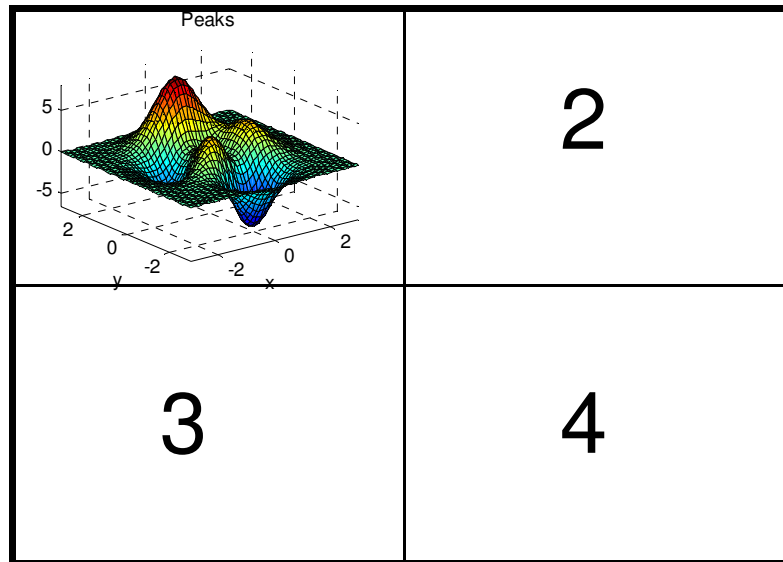
- Grafik penceresini m satır, n sütun olacak şekilde ölçeklendirerek bölümlere ayırmak ve birden fazla grafik eklemek mümkün.

subplot(m,n,p)

satır sütun konum

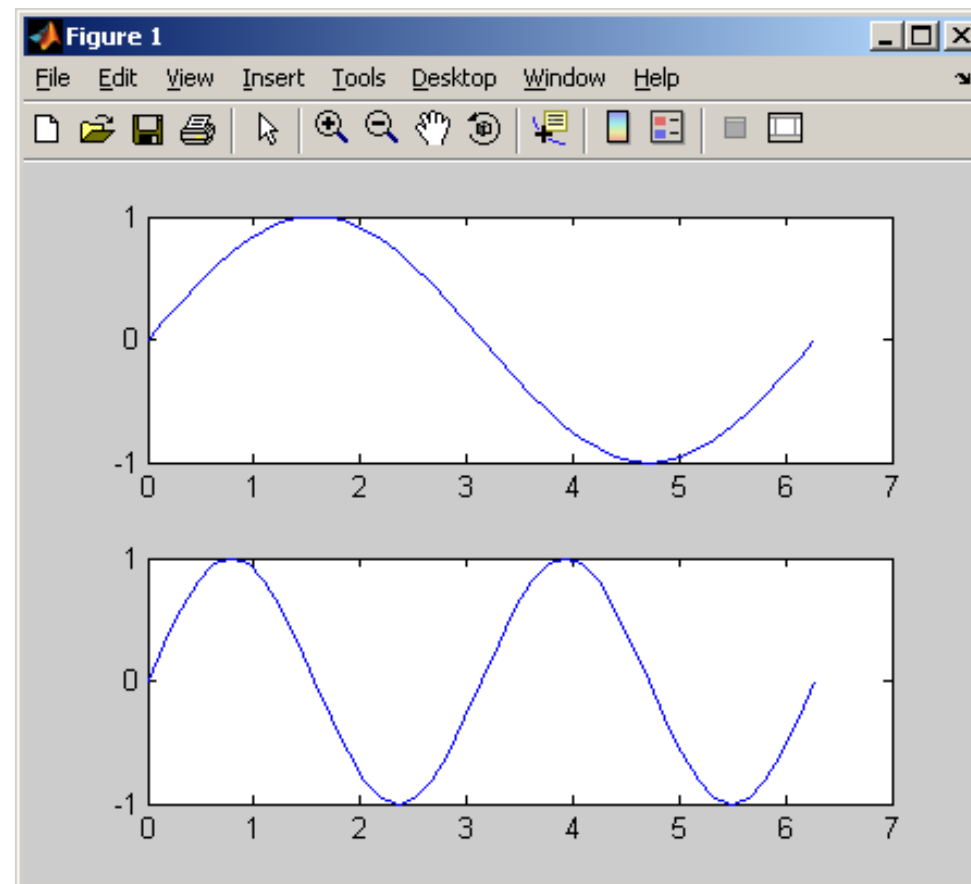
2 sütun

2 satır



```
Command Window
File Edit Debug Desktop Window Help

>> x=0:pi/20:2*pi;
>> subplot(2,1,1)
>> plot(x,sin(x))
>> subplot(2,1,2)
>> plot(x,sin(2*x))
>> |
```



2 BOYUTLU ÇİZİMLERİN DİĞER TÜRLERİ

- Kutupsal çizimler (Polar Plots)
- Logaritmik çizimler
- Sütun (çubuk) grafiği (Bar Graphs)
- Pasta grafiği (Pie Charts)
- Histogramlar
- İki y eksenli x-y grafiği
- Fonksiyon çizimleri

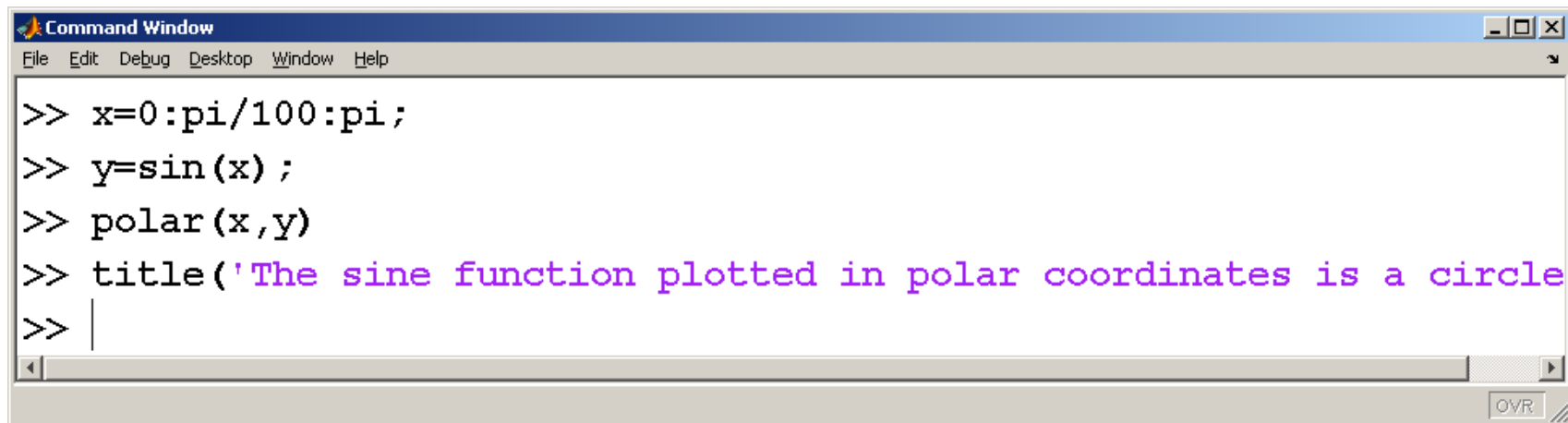


Kutupsal Çizimler

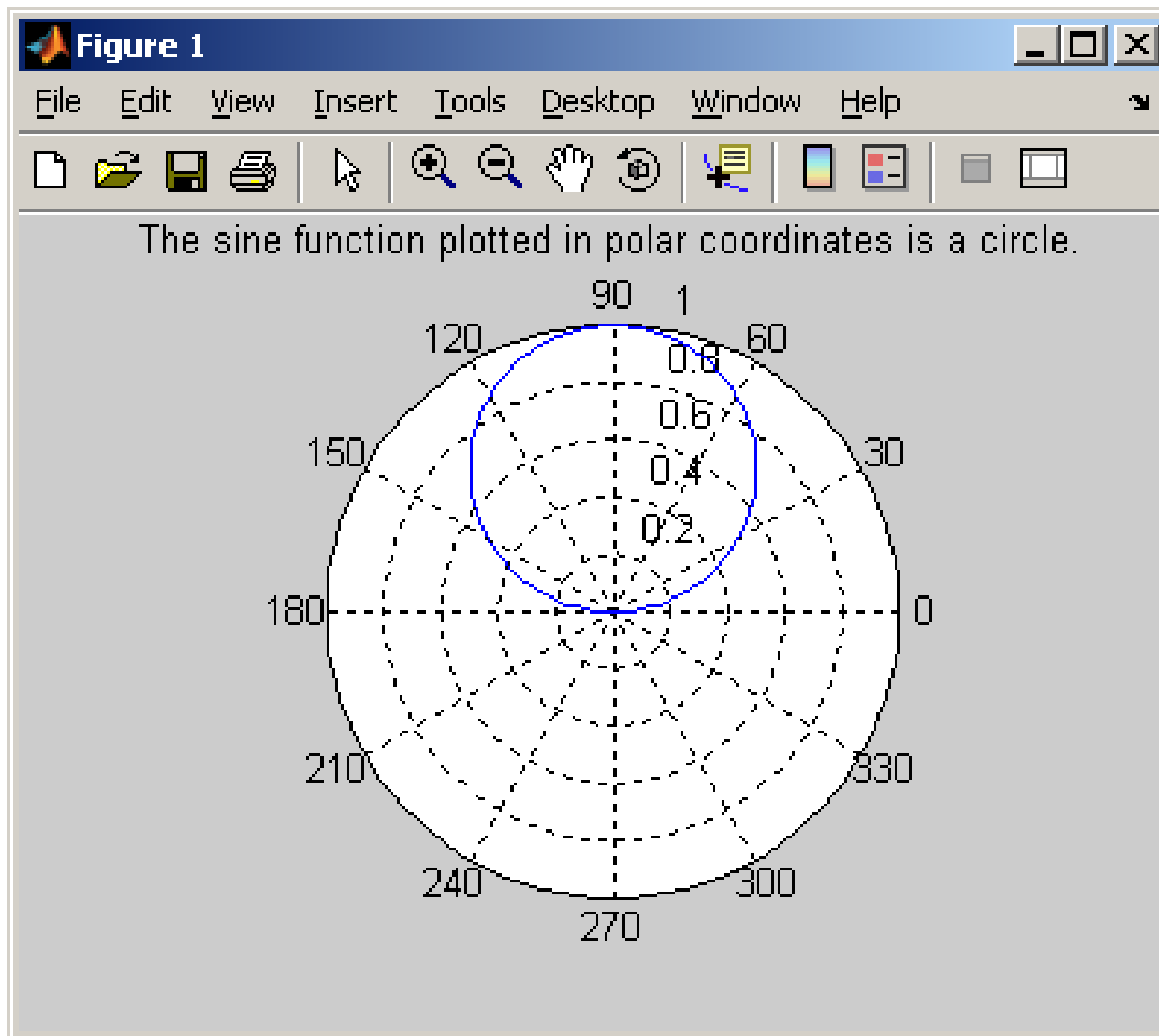
Bazı fonksiyonları dik koordinatlar kullanmak yerine kutupsal koordinatlar kullanarak çizdirmek daha kolaydır.

Örneğin çemberin denklemini kutupsal koordinatlarla çizdirelim:

$$y = \sin(x)$$



```
Command Window
File Edit Debug Desktop Window Help
>> x=0:pi/100:pi;
>> y=sin(x);
>> polar(x,y)
>> title('The sine function plotted in polar coordinates is a circle')
>> |
```



Logaritmik Çizimler

Logaritmik ölçeklendirme (10 tabanında)

- değişken aralıkları çok geniş olduğunda kullanılır çünkü büyük aralıklı değerler daha küçük değerleri sıkıştırmadan çizilebilir.
- veriler üstel olarak değiştiğinde kullanılır.

plot → Her iki ekseninde lineer ölçekleme yapar

semilogy → y ekseninde \log_{10} ölçekleme yapar

semilogx → x ekseninde \log_{10} ölçekleme yapar

loglog → Her iki ekseninde \log_{10} ölçekleme yapar



```
1      x = 0:0.5:50;
2      y = 5*x.^2;
3      subplot(2,2,1)
4      plot(x,y)
5          title('Polynomial - linear/linear')
6          ylabel('y'), grid
7      subplot(2,2,2)
8      semilogx(x,y)
9          title('Polynomial - log/linear')
10         ylabel('y'), grid
11      subplot(2,2,3)
12      semilogy(x,y)
13         title('Polynomial - linear/log')
14         xlabel('x'), ylabel('y'), grid
15      subplot(2,2,4)
16      loglog(x,y)
17         title('Polynomial - log/log')
18         xlabel('x'), ylabel('y'), grid
```

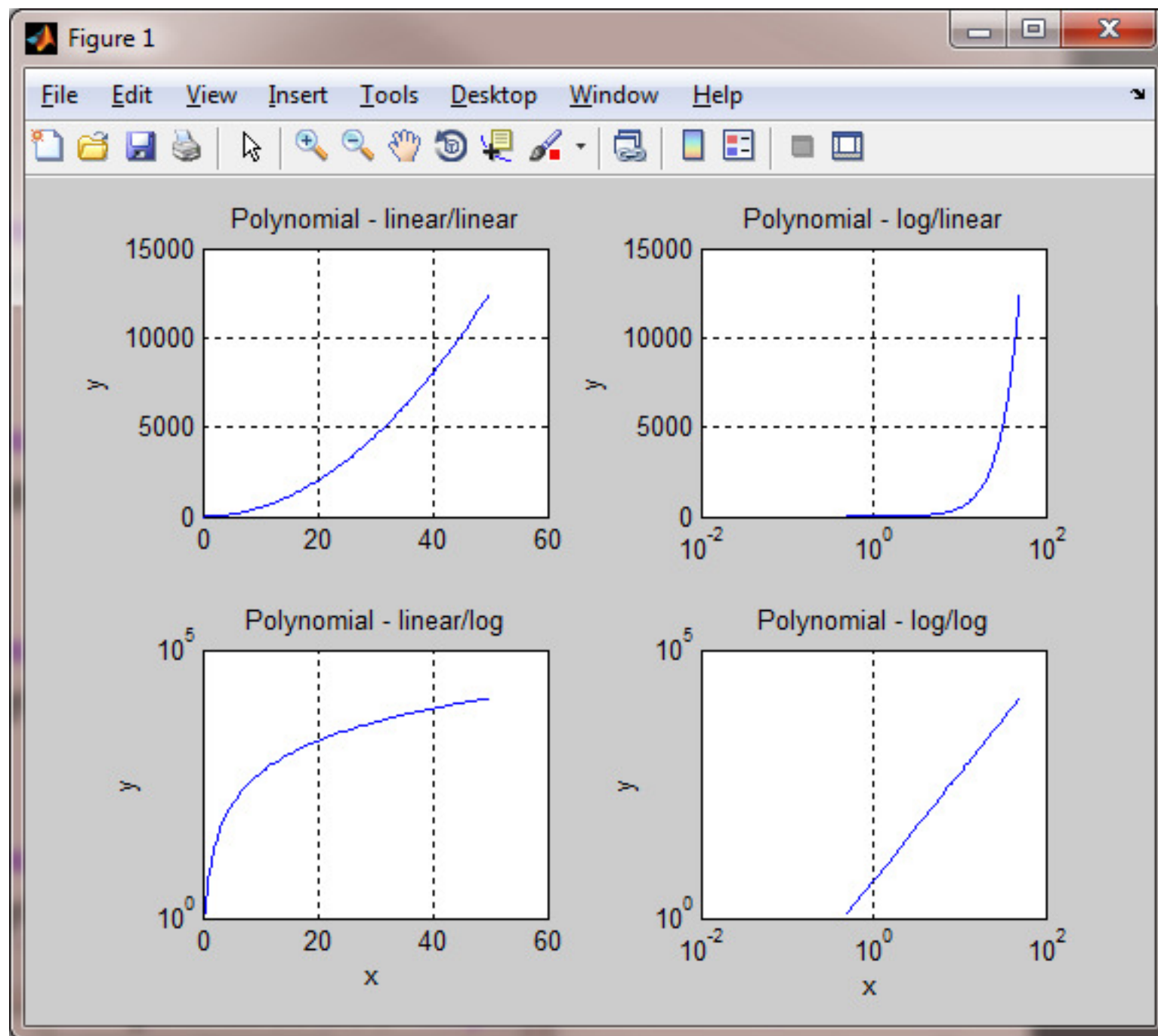
x-y plot – her iki
eksen lineer

semilogx – x ekseninde
log ölçekleme

semilogy –y ekseninde log
ölçekleme

loglog – her iki ekseninde log
ölçekleme

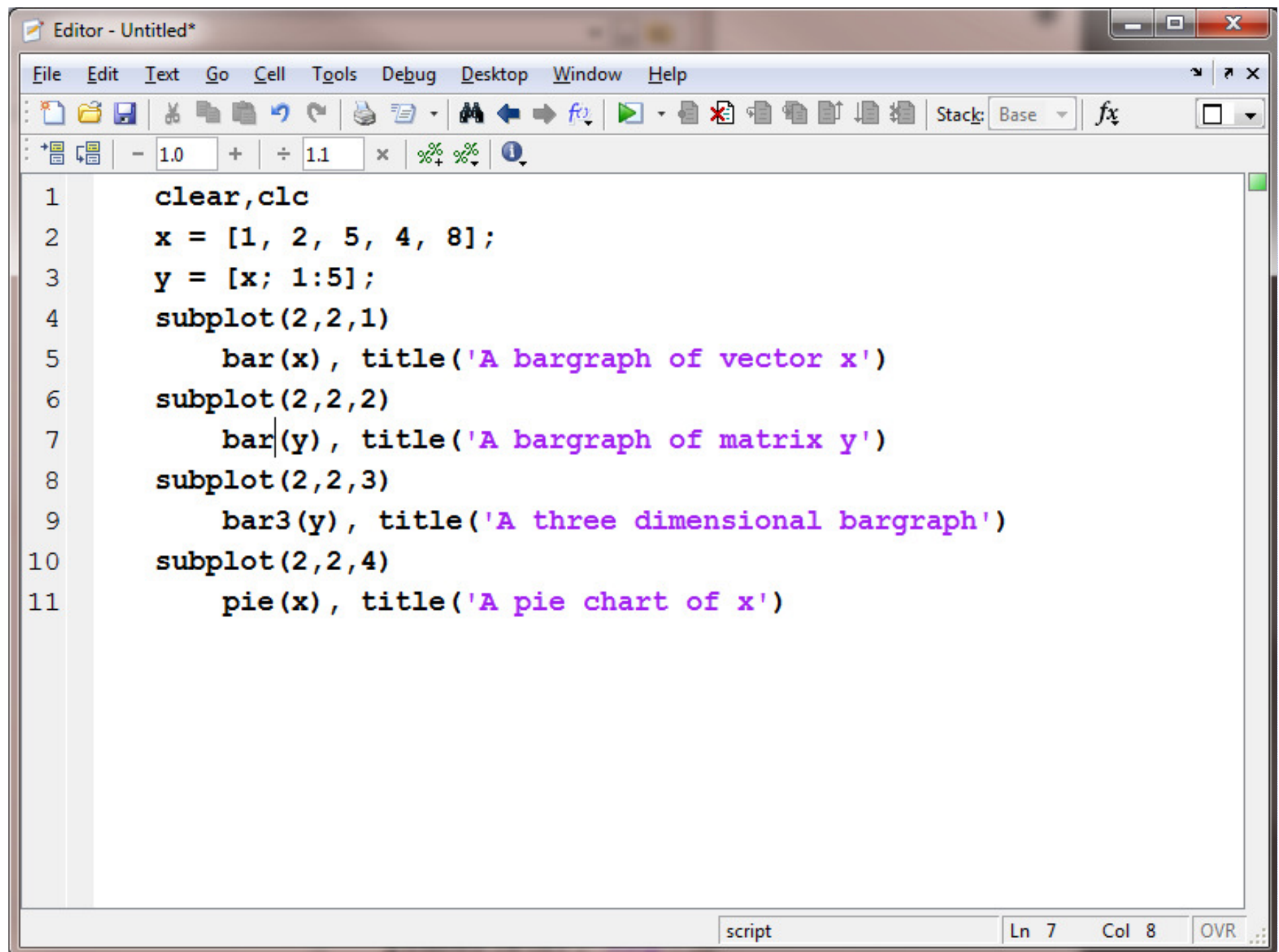
script Ln 18 Col 35 OVR



Sütun ve Pasta Grafikleri

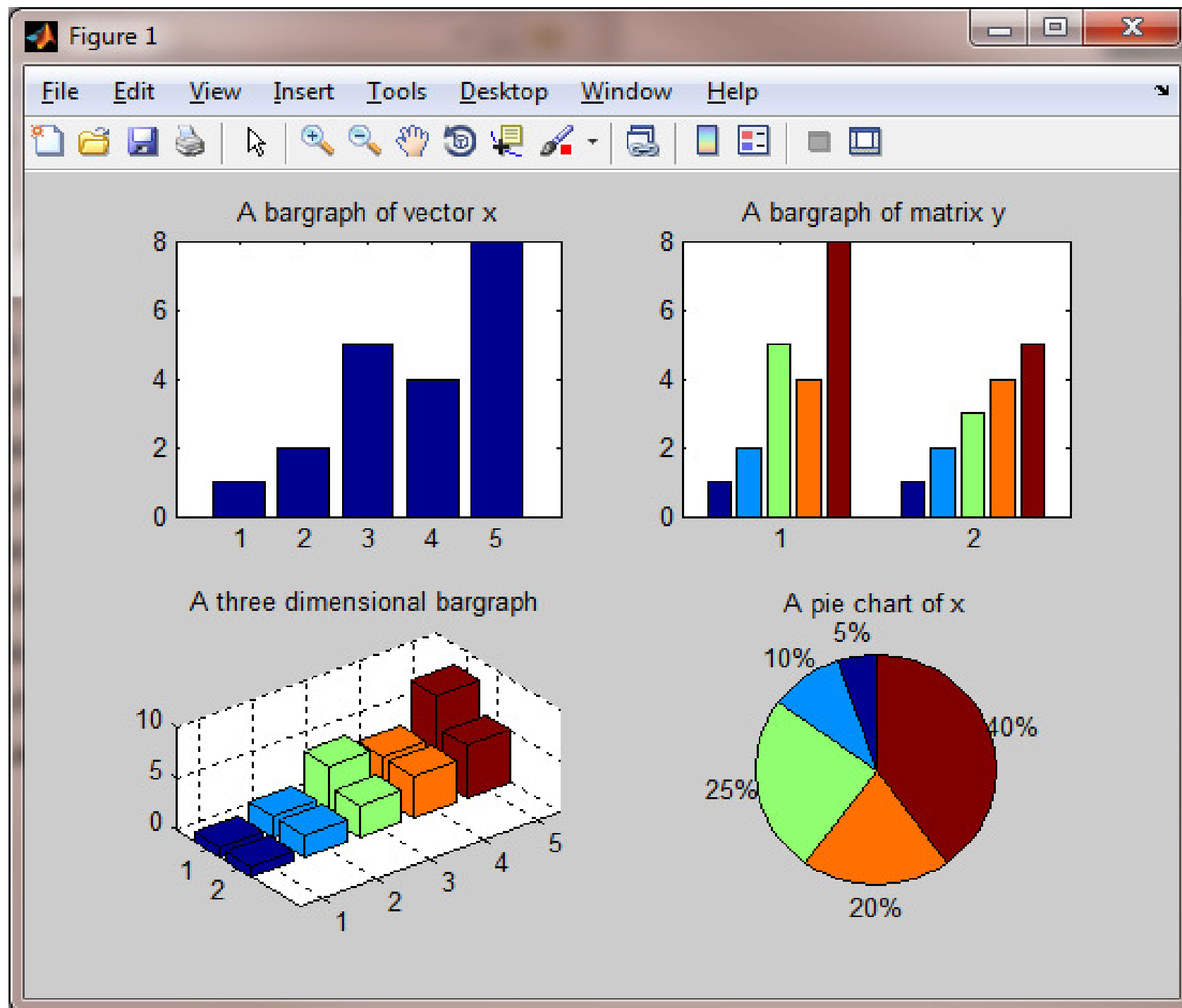
- `bar(x)` – dikey sütun grafiği
- `barh(x)` – yatay sütun grafiği
- `bar3(x)` – 3-D dikey sütun grafiği
- `bar3h(x)` – 3-D yatay sütun grafiği
- `pie(x)` – pasta grafiği
- `pie3(x)` – 3-D pasta grafiği





The image shows a MATLAB Editor window titled "Editor - Untitled*". The window has a menu bar with "File", "Edit", "Text", "Go", "Cell", "Tools", "Debug", "Desktop", "Window", and "Help". Below the menu bar is a toolbar with various icons for file operations, editing, and execution. The main editing area contains a script with 11 lines of MATLAB code. The code defines a vector x, a matrix y, and creates four subplots: a bar graph of x, a bar graph of matrix y, a 3D bar graph of y, and a pie chart of x. The status bar at the bottom indicates the current position is at line 7, column 8, and the file is named "script".

```
1 clear,clc
2 x = [1, 2, 5, 4, 8];
3 y = [x; 1:5];
4 subplot(2,2,1)
5     bar(x), title('A bargraph of vector x')
6 subplot(2,2,2)
7     bar(y), title('A bargraph of matrix y')
8 subplot(2,2,3)
9     bar3(y), title('A three dimensional bargraph')
10 subplot(2,2,4)
11     pie(x), title('A pie chart of x')
```

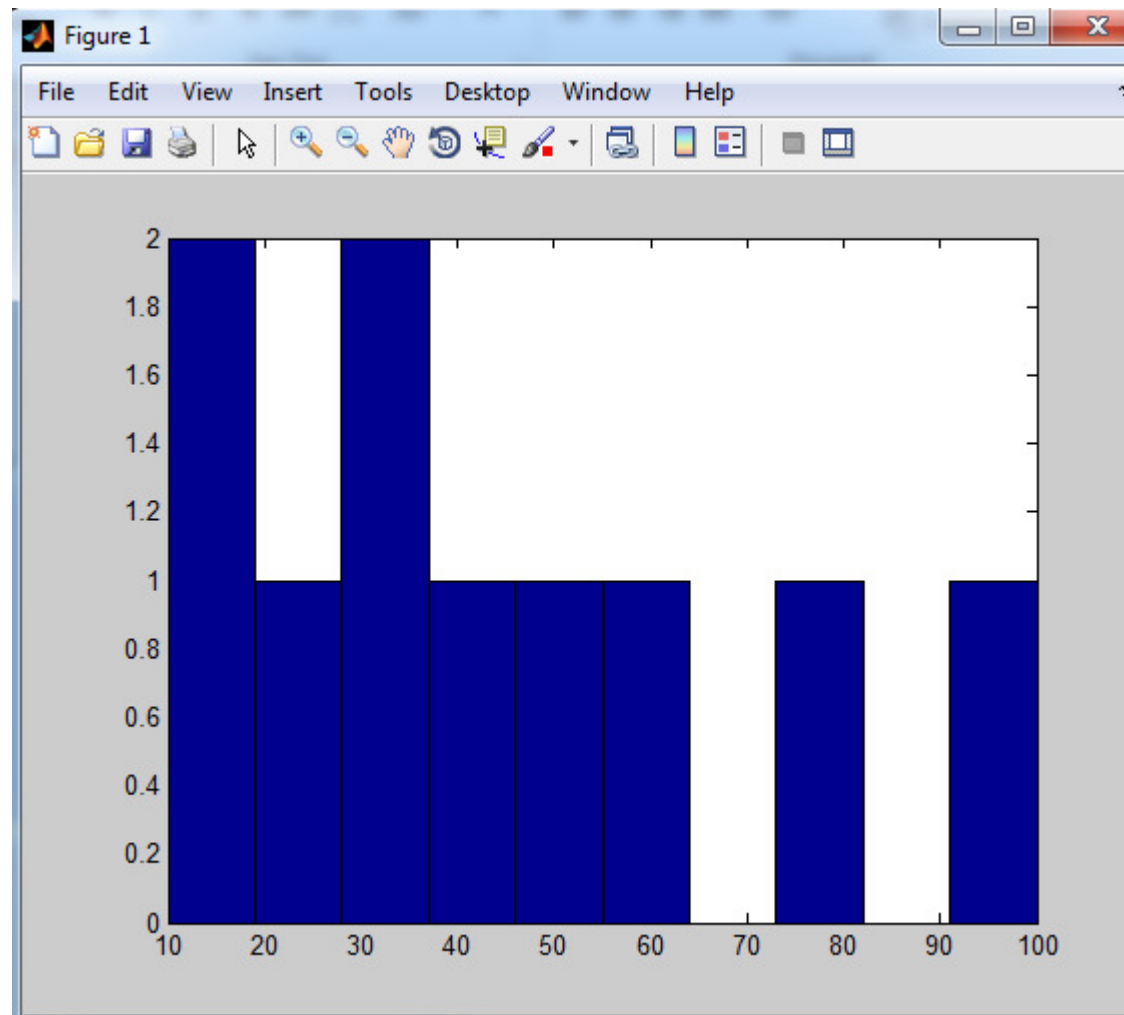



Histogram

- Histogram bir kümedeki değerlerin dağılımını gösteren çizimdir.

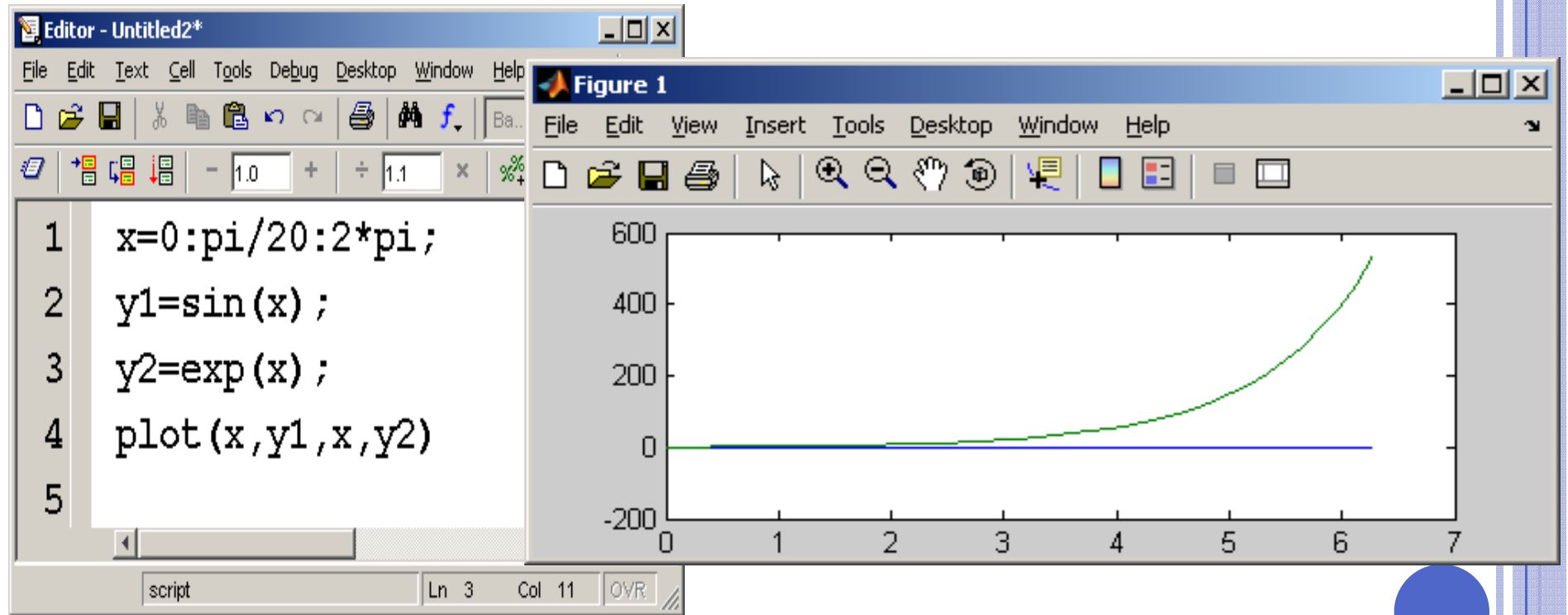
```
>> x=[20,30,10,40,60,50,10,30,80,100];
```

```
>> hist(x)
```



İki y Eksenli x-y Grafiği

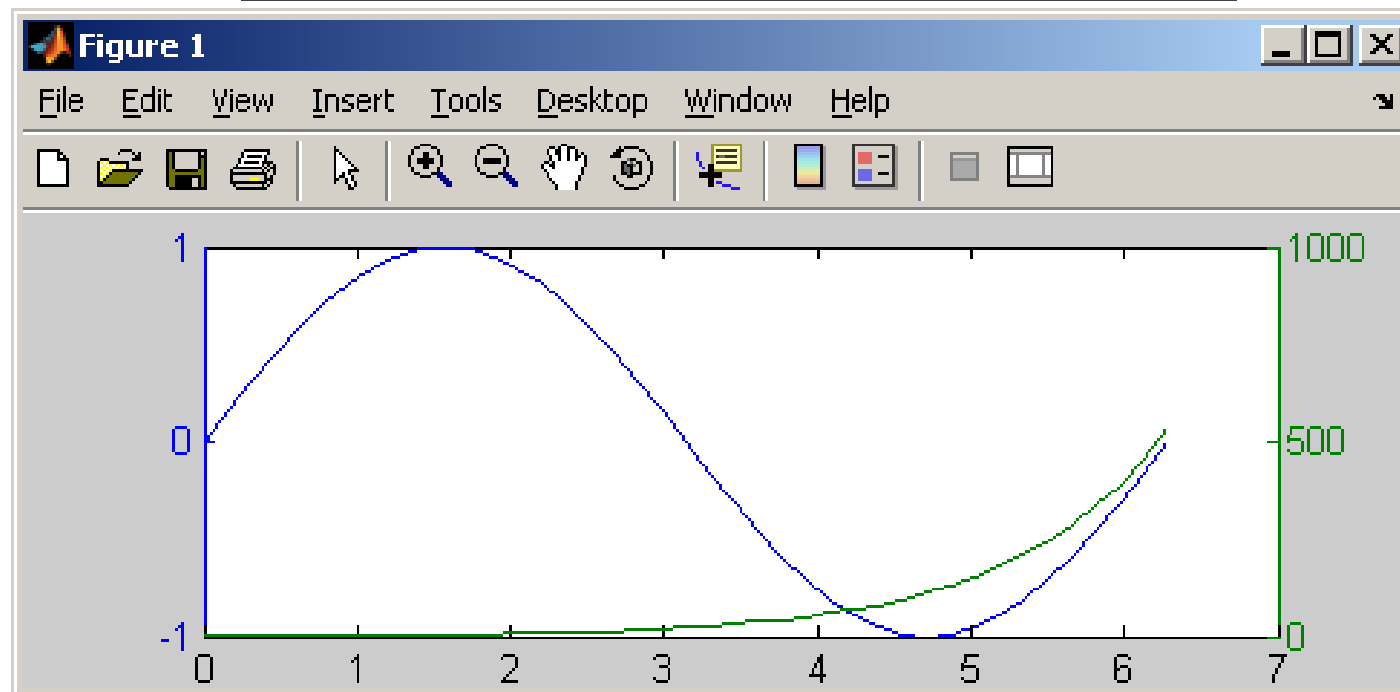
Bazen iki x-y grafiğini aynı şekilde göstermek kullanışlı olmaktadır. Fakat y değerlerinin büyüklük sırası (genliği) birbirinden çok farklıysa, verinin nasıl davrandığını görmek zor olabilir.



Mavi çizginin ölçeği uygun olmadığı için nasıl davrandığını anlamak zor olmaktadır.

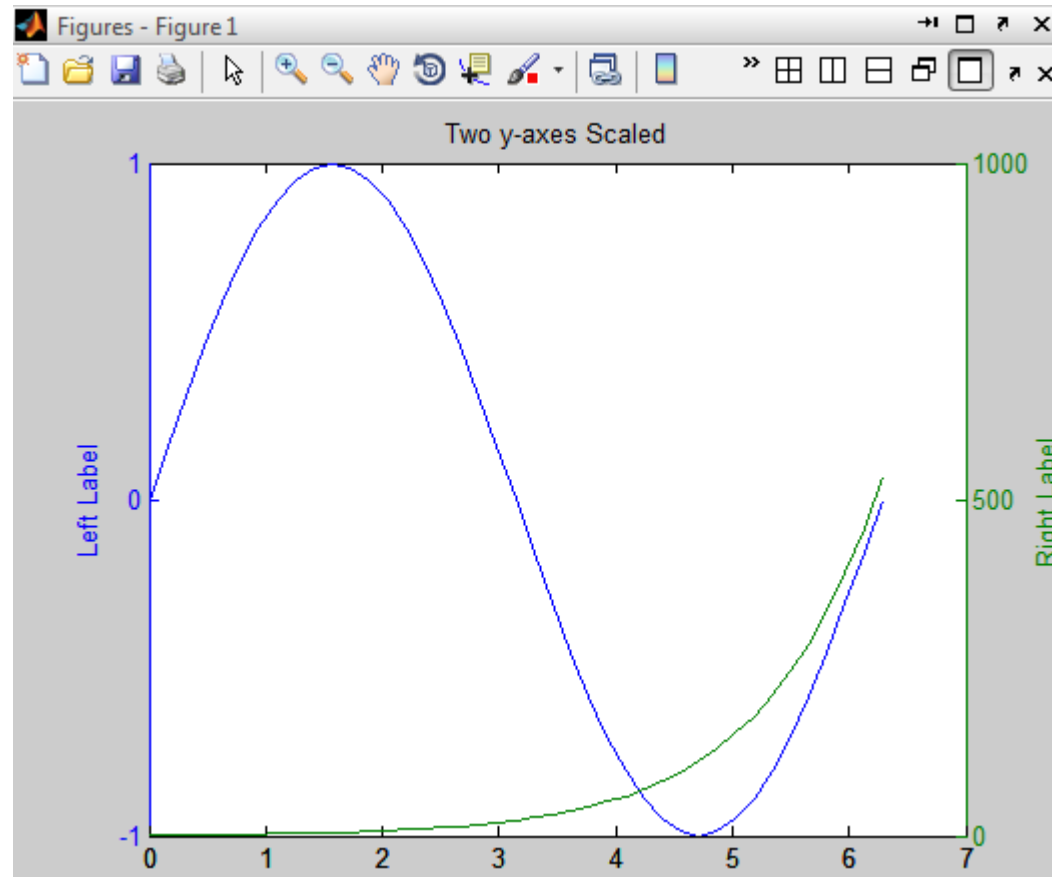
plotyy
fonksiyonu
tek bir grafik
üzerinde iki
ölçeklendirme
yapmanızı
sağlar.

```
Editor - Untitled2*
File Edit Text Cell Tools Debug Desktop Window Help
[Icons] Ba... [Grid]
[Icons] [Zoom In] [Zoom Out] [Zoom Reset] [Zoom Extends] [Zoom Full]
1 x=0:pi/20:2*pi;
2 y1=sin(x);
3 y2=exp(x);
4 plotyy(x,y1,x,y2)
5
script Ln 2 Col 11 OVR
```



```
Command Window

>> clear
>> x=0:pi/20:2*pi;
>> y1=sin(x);
>> y2=exp(x);
>> a=plotyy(x,y1,x,y2);
>> title('Two y-axes Scaled')
>> ylabel('Left Label')
>> ylabel(a(2), 'Right Label')
>>
```



Fonksiyon Çizimleri

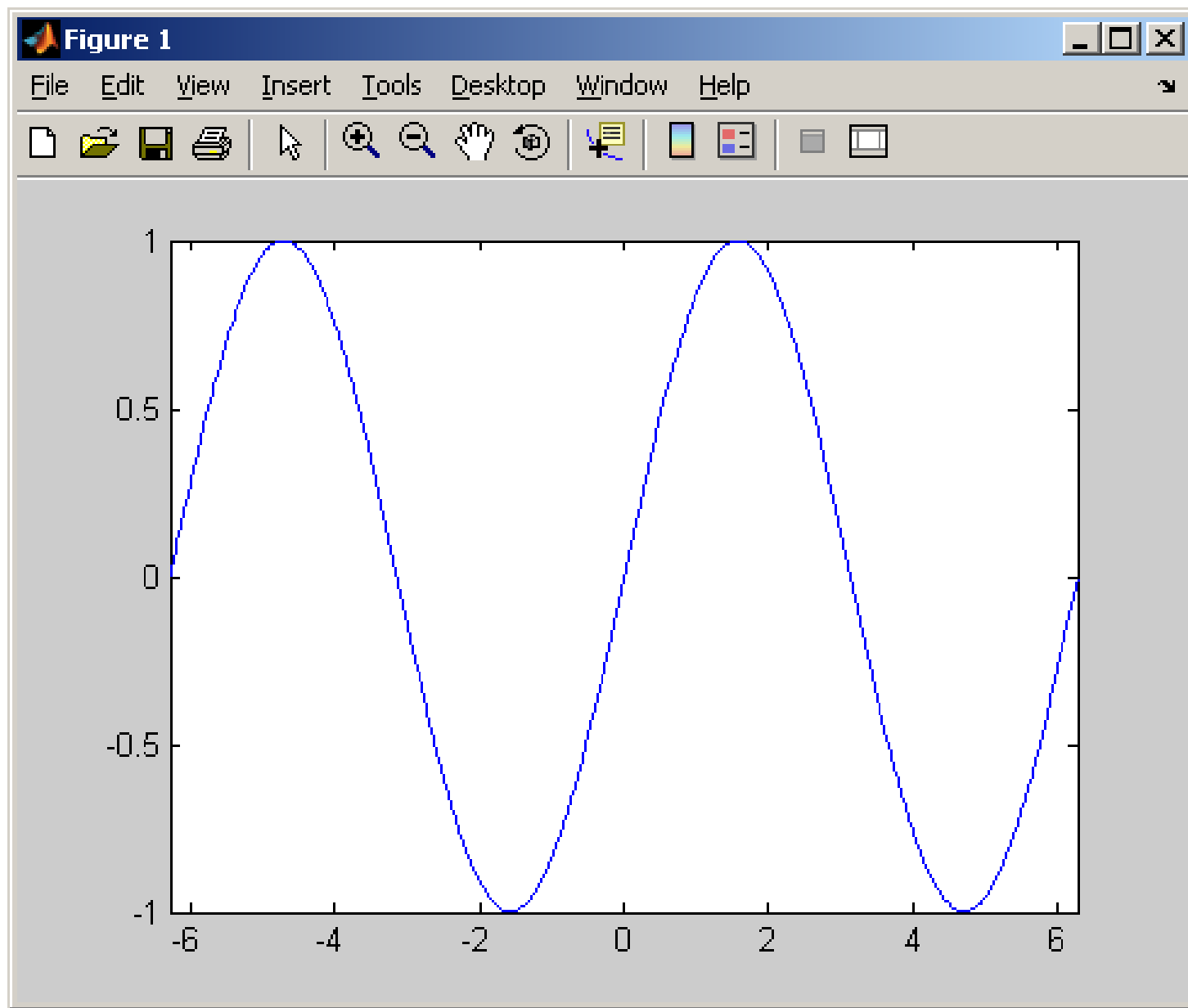
Fonksiyon çizimleri, input olarak sıralı x-y değerlerini girmek yerine plot komutuna input olarak bir fonksiyon girmemizi sağlar.

`fplot ('sin(x)',[-2*pi,2*pi])`

Fonksiyon girdisi

x değişkeninin aralıkları





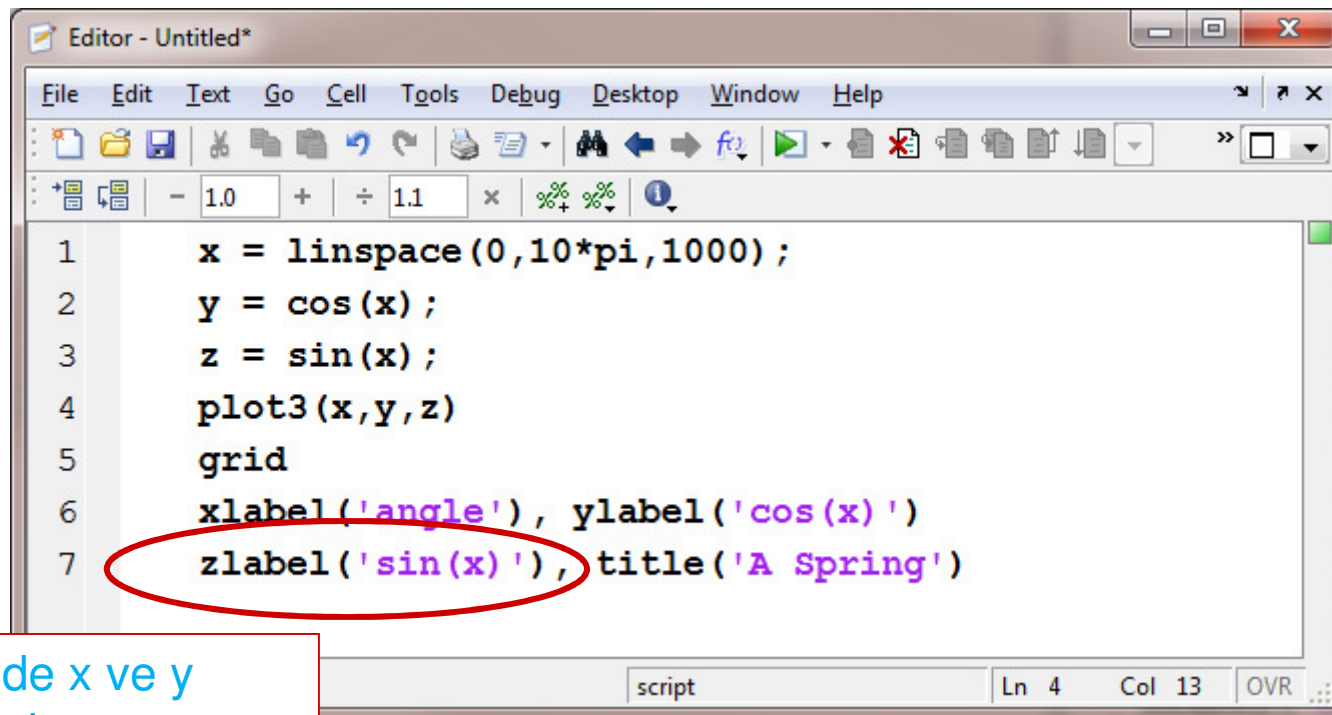
Üç Boyutlu Çizim

- Line plots (Çizgi grafiği)
- Surface plots (Yüzey grafiği)
- Contour plots (Kenar grafiği)



Üç Boyutlu Çizgi Grafiği

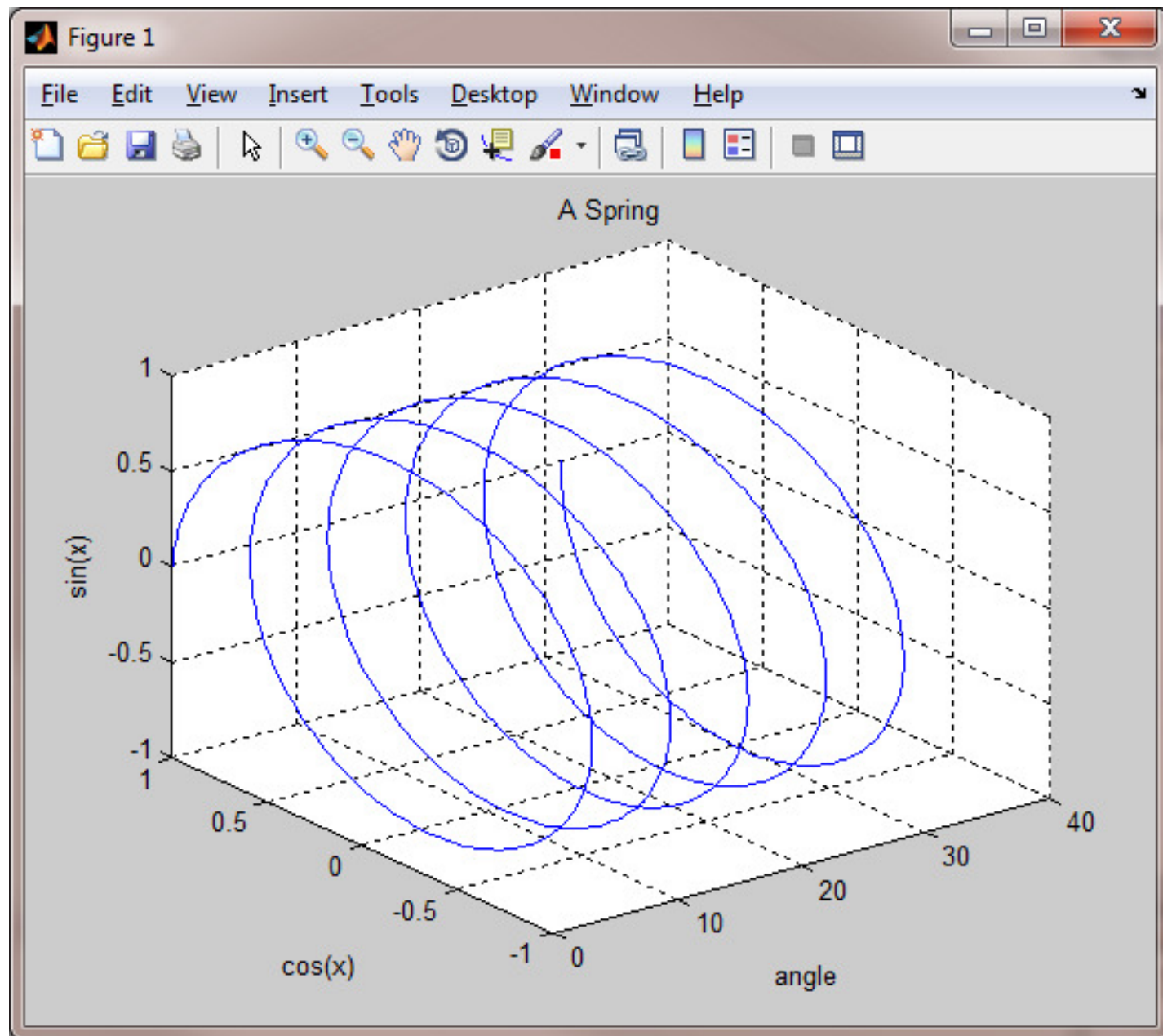
Bu grafikler girdi olarak üçlü değere (x-y-z değerleri) ihtiyaç duyarlar.



```
1 x = linspace(0,10*pi,1000);  
2 y = cos(x);  
3 z = sin(x);  
4 plot3(x,y,z)  
5 grid  
6 xlabel('angle'), ylabel('cos(x)')  
7 zlabel('sin(x)'), title('A Spring')
```

z-ekseni de x ve y
eksenlerinin
etiketlendiği gibi
etiketlenir.





comet3 fonksiyonuyla grafiği animasyon şeklinde çizdirebilirsiniz.

comet3(x,y,z)

2-D çizgi grafikleri için **comet** fonksiyonunu kullanabilirsiniz.



Yüzey Grafiği

x-y-z verisini yüzey olarak gösterir

mesh - meshplot

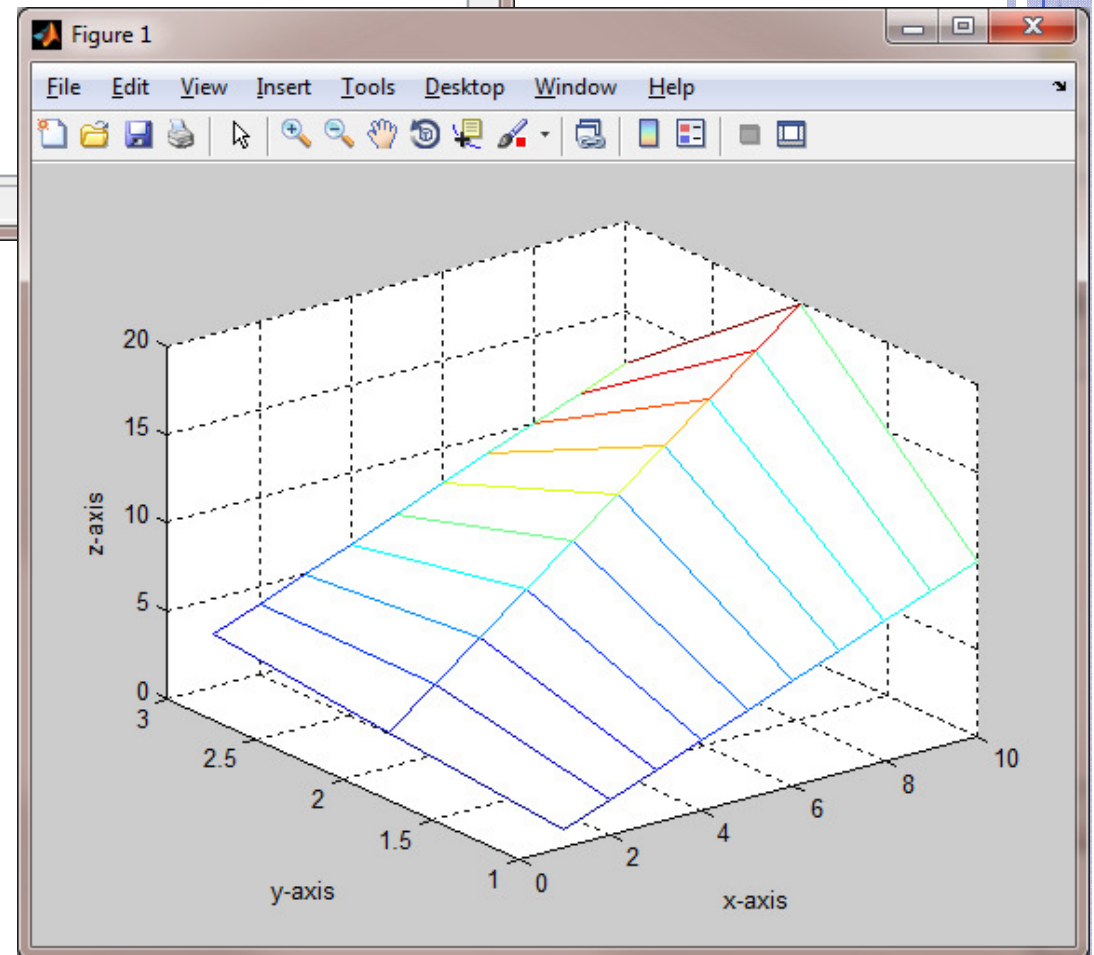
surf - surface plot

Her iki fonksiyon da tek bir iki boyutlu matrisle kullanılabilir.



```
Editor - Untitled*
File Edit Text Go Cell Tools Debug Desktop Window Help
1 z = [1 2 3 4 5 6 7 8 9 10;
2     2 4 6 8 10 12 14 16 18 20;
3     3 4 5 6 7 8 9 10 11 12];
4 mesh(z)
5 xlabel('x-axis')
6 ylabel('y-axis')
7 zlabel('z-axis')
script
```

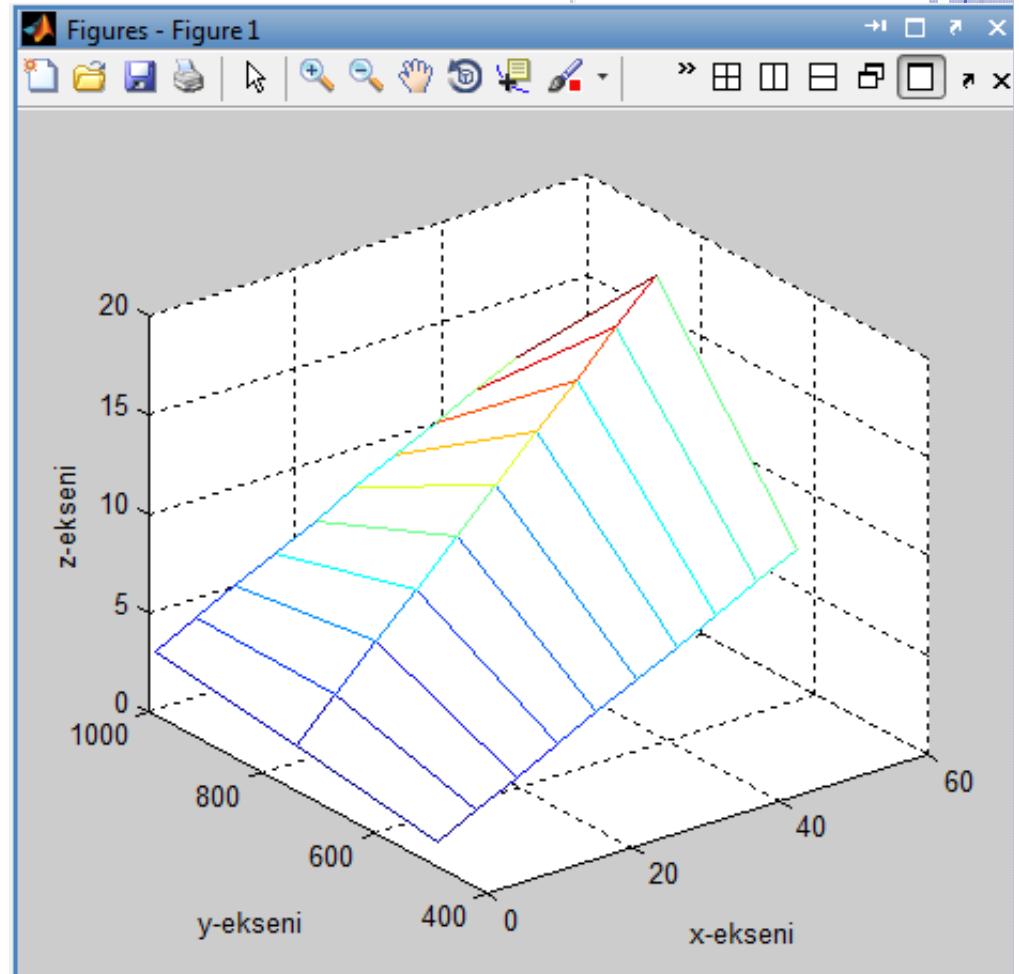
x ve y koordinatları
matrisin index numaraları



x ve y değerlerini biliyorsak index numaraları yerine o değerlere göre çizim yaptırabiliriz.

Command Window

```
>> z=[1:10; 2:2:20; 3:12];  
>> x=linspace(1,50,10);  
>> y=linspace(500,1000,3);  
>> mesh(x,y,z)  
>> xlabel('x-ekseni')  
>> ylabel('y-ekseni')  
>> zlabel('z-ekseni')
```



surf fonksiyonu da mesh fonksiyonuna benzerdir

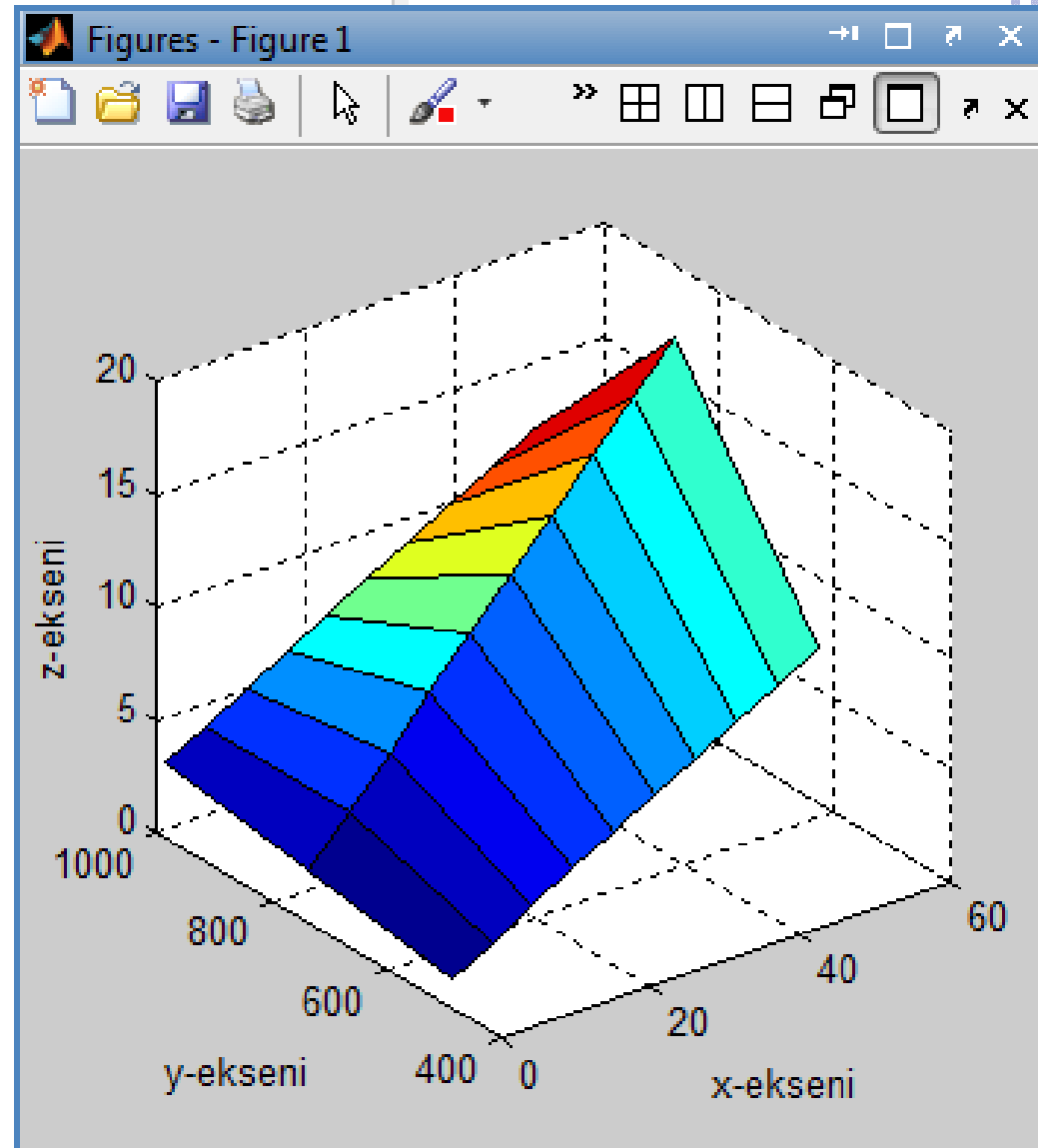
- Açık ızgara (mesh) yerine 3-D renkli yüzey oluşturur.
- Yapısı aynıdır.



Command Window

 New to MATLAB? Watch this [Video](#), see [Demos](#), or read [Get](#) X

```
>> z=[1:10; 2:2:20; 3:12];  
>> x=linspace(1,50,10);  
>> y=linspace(500,1000,3);  
>> surf(x,y,z)  
>> xlabel('x-ekseni')  
>> ylabel('y-ekseni')  
>> zlabel('z-ekseni')
```

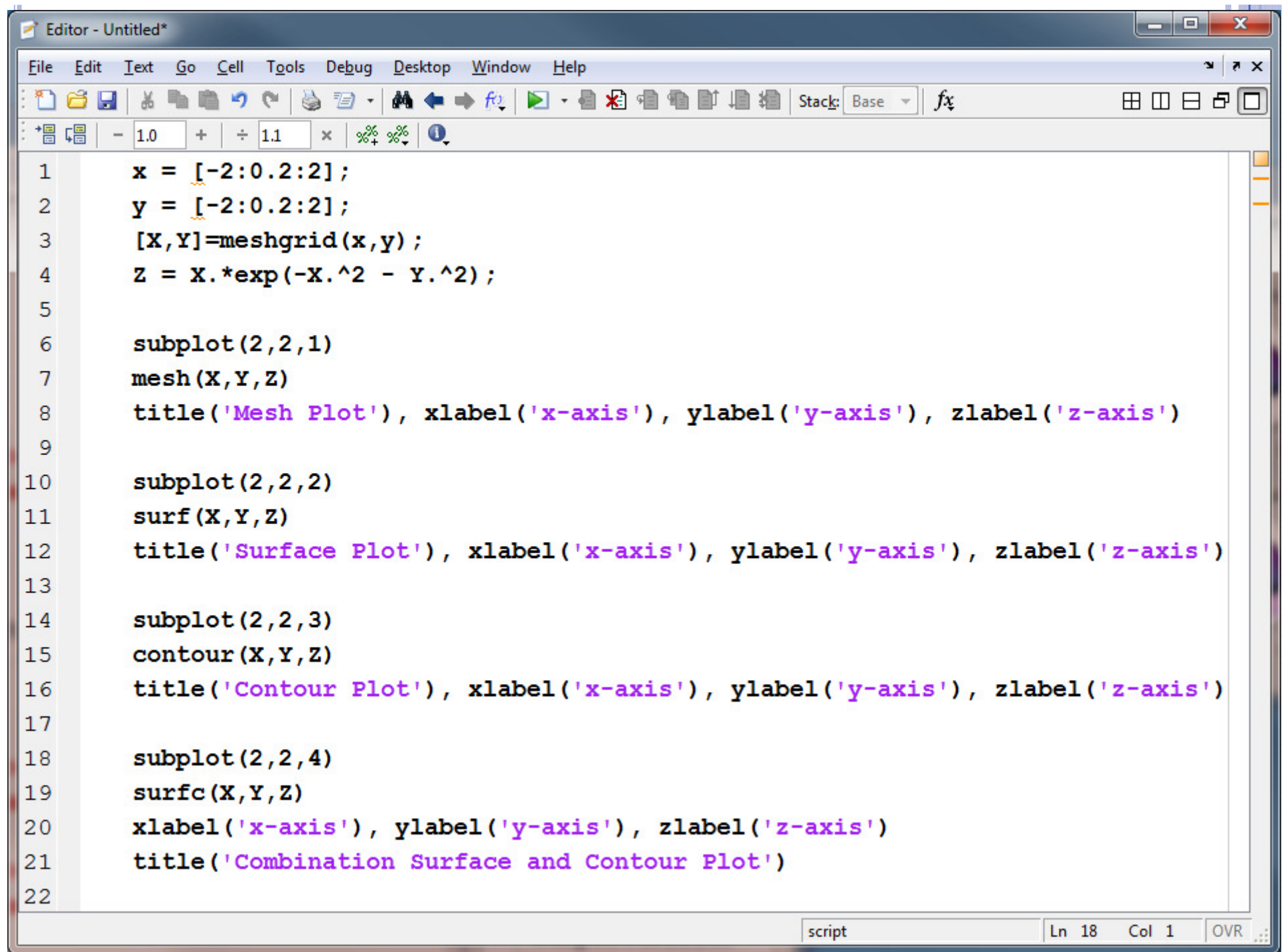


Kenar Grafiği (Contour Plots)

Contour plot da mesh ve surface plotlarında kullanılan şekilde input bilgisini alır.

Uzun yürüyüş yapan kişilerin (hikers) kullandığı eşyüksekti haritasına benzer yapıda grafik oluşturur.





The image shows a MATLAB Editor window titled "Editor - Untitled*". The window contains a script with 22 lines of MATLAB code. The code defines a 2D grid (X, Y) and calculates a Z-value for each point using the formula $Z = X \cdot \exp(-X^2 - Y^2)$. It then creates four subplots: a mesh plot, a surface plot, a contour plot, and a combination surface and contour plot. Each plot is titled and labeled with axes.

```
1  x = [-2:0.2:2];
2  y = [-2:0.2:2];
3  [X,Y]=meshgrid(x,y);
4  Z = X.*exp(-X.^2 - Y.^2);
5
6  subplot(2,2,1)
7  mesh(X,Y,Z)
8  title('Mesh Plot'), xlabel('x-axis'), ylabel('y-axis'), zlabel('z-axis')
9
10 subplot(2,2,2)
11 surf(X,Y,Z)
12 title('Surface Plot'), xlabel('x-axis'), ylabel('y-axis'), zlabel('z-axis')
13
14 subplot(2,2,3)
15 contour(X,Y,Z)
16 title('Contour Plot'), xlabel('x-axis'), ylabel('y-axis'), zlabel('z-axis')
17
18 subplot(2,2,4)
19 surfc(X,Y,Z)
20 xlabel('x-axis'), ylabel('y-axis'), zlabel('z-axis')
21 title('Combination Surface and Contour Plot')
22
```

The status bar at the bottom indicates the current position is at line 18, column 1, and the file is named "script".

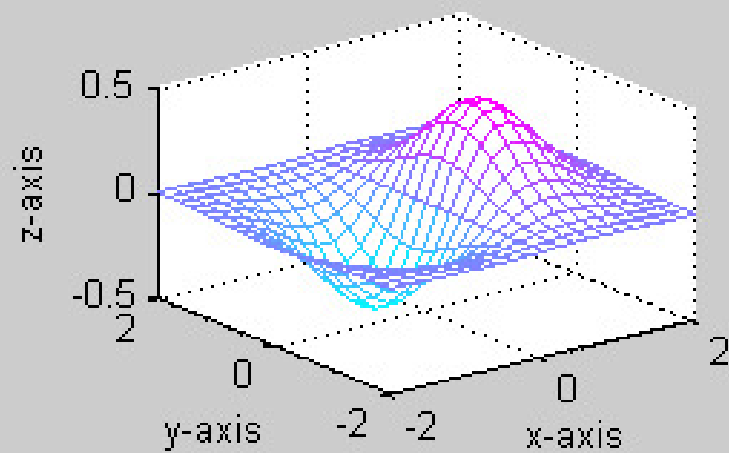


Figure 1

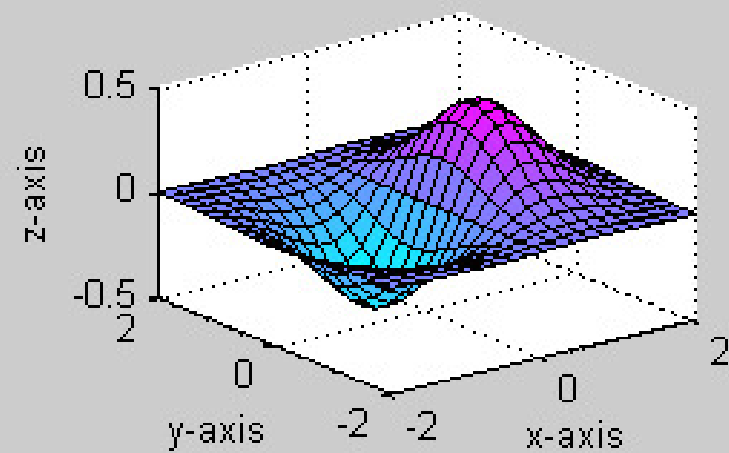
File Edit View Insert Tools Desktop Window Help



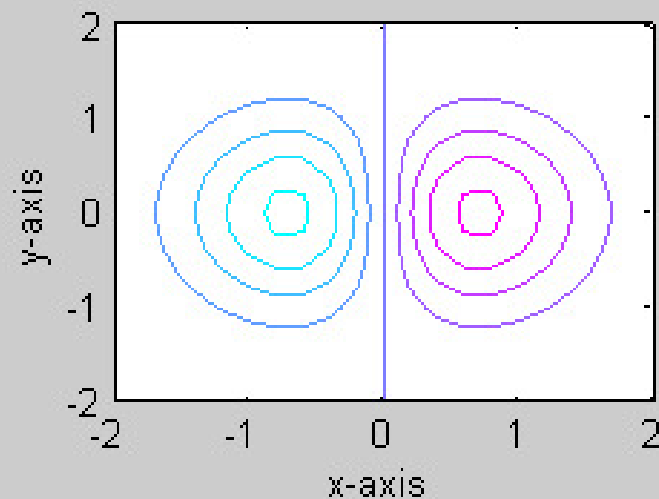
Mesh Plot



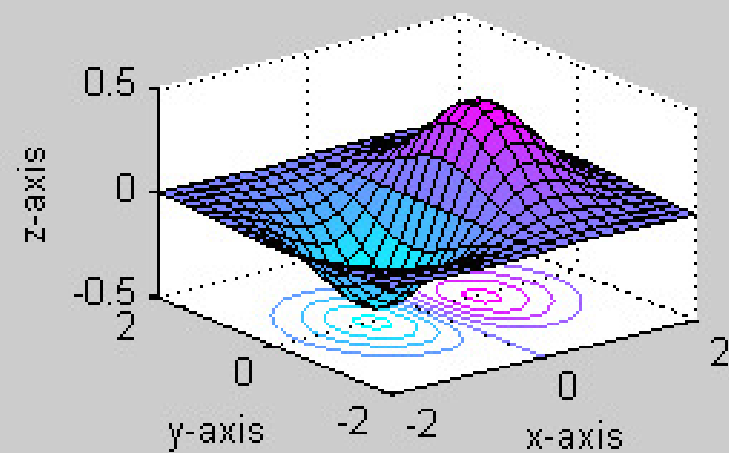
Surface Plot



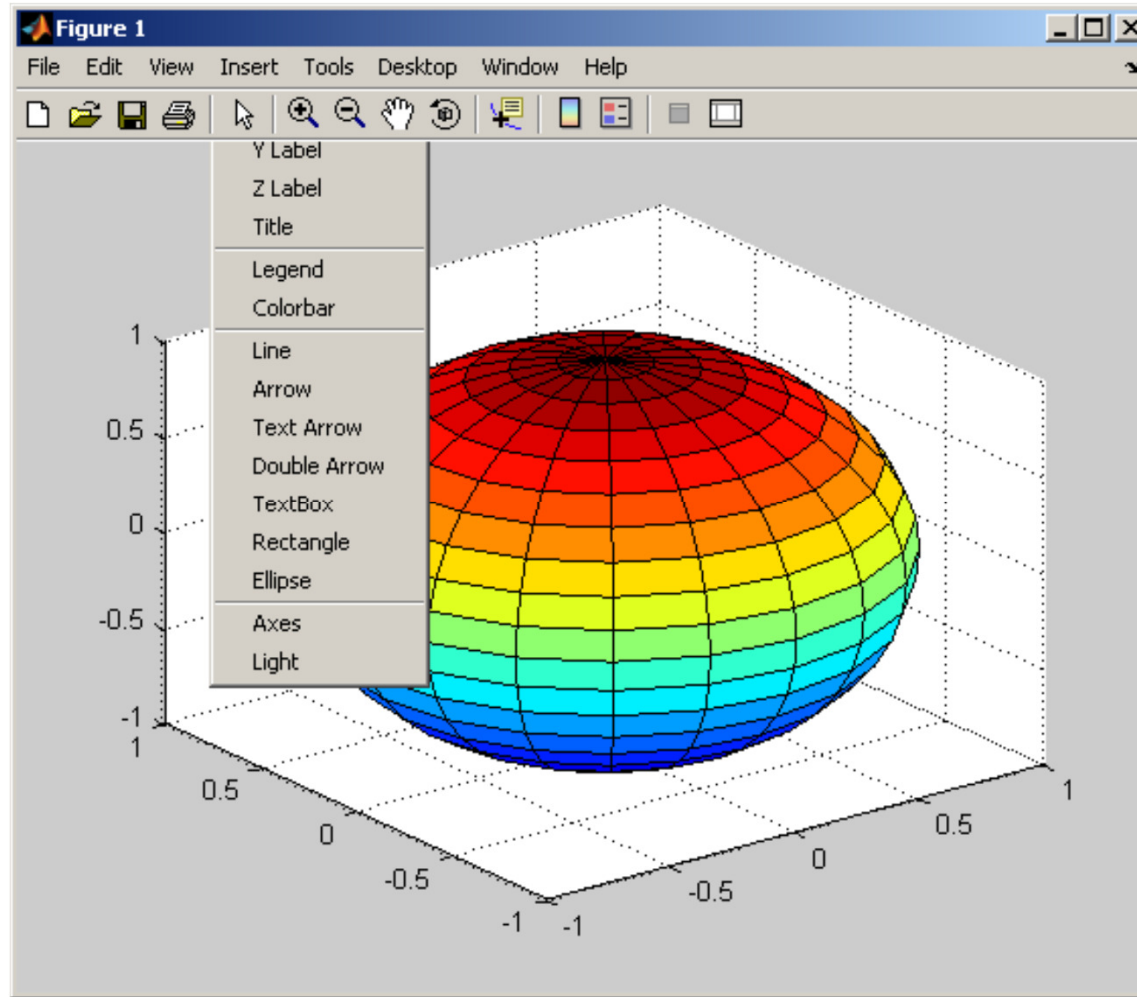
Contour Plot



Combination Surface and Contour Plot



Grafik oluřturduktan sonra Menü ubuęunda Insert menüsünü kullanarak grafikte dzenlemeler yapabilirsiniz.



**Oluřturduėunuz grafiėi File menüsünden
Save As seçeneėi ile farklı formatlarda
kaydedebilirsiniz.**

