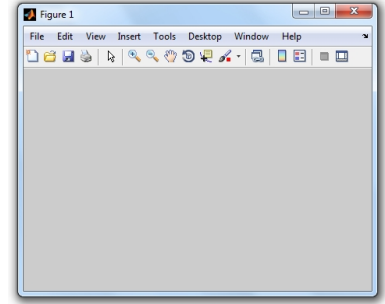


2 Boyutlu Grafikler

Giriş

2 boyutlu bilgisayar grafikleri koordinat düzleminde x ve y eksenlerine karşılık gelen değerlerin incelenmesiyle oluşurlar. Matlab çok çeşitli grafik çizme yeteneklerine sahiptir. Grafik çizim komutları hem komut satırında hem M-dosyalarında kullanılabilir. Geliştirilmiş bir çok fonksiyon sayesinde çizilen grafikler özelleştirilebilir. Bu bölümde 2 boyutlu grafikler çizmek için geliştirilmiş olan fonksiyonları inceleyeceğiz.

Matlab herhangi bir anda herhangi bir fonksiyondan grafik çizme emri aldığı anda bir figür (figure) nesnesi oluşturur. Bu nesne bir veya daha fazla grafik eksenini barındırabileceği gibi gösterilen grafikler üzerinde dönme, yaklaşma, uzaklaşma, kaydetme vb. birçok niteliğe erişimi sağlayan araç çubuklarına sahiptir.



Şekil 3: Bir figür nesnesi, matlab tarafından çizilen tüm grafikler için ev sahipliği yapar.

plot fonksiyonu

Matlab'da 2 boyutlu grafikler çizmek için kullanılan temel fonksiyon **plot** fonksiyonudur. Grafiği çizilecek olan matematiksel fonksiyonun x değişkeninin her değeri için hesaplatılan grafiğe ait (x, y) noktalarının koordinat düzleminde işaretlenmesiyle grafik oluşturulur. x ve y değerlerinin oluşturacağı dizilerin aynı uzunlukta olmasına dikkat edilmelidir. Aksi halde grafik çizilmeyecektir. **plot** fonksiyonunun en yalın halinde, giriş parametresi olarak x ve y dizileri kullanılır.

Örnek 57 $y = x^3 - 3x^2 + 7x + 5$ fonksiyonunun grafiğini x değerleri $[-5, 5]$ aralığında olacak şekilde çiziniz.

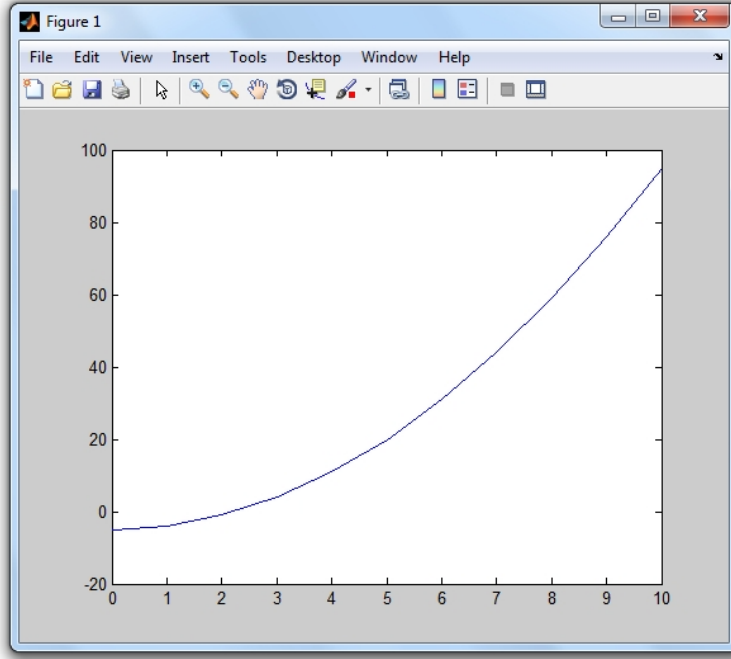
y fonksiyonunun grafiğinin çizilebilmesi için ilk olarak x ve y nin alabileceği değerler dizisi oluştururlur.

```
x = 0:1:10
y = x.^2 - 5
plot(x, y)
```

```
>>
```

```
x = 0    1    2    3    4    5    6    7    8    9   10
y = -5   -4   -1    4   11   20   31   44   59   76   95
```

Yukarıdaki gibi eşit sayıda elemana sahip x ve y dizileri için `plot` fonksiyonu uygulandığında ilk olarak figür nesnesi oluşturulur. Daha sonra, eksen nesnesi oluşturulur ve içerisinde grafik çizilir.



Şekil 4: x ve y dizilerinin oluşturduğu (x, y) ikilileri, grafiğin kesim noktalarını oluşturmaktadır.

Figür nesnesi üzerindeki araç çubukları sayesinde grafiğe farklı açılardan bakmak, resim olarak kaydetmek, yazdırmak gibi birçok işlem gerçekleştirilir.





Not 6 2 boyutlu bir grafiği biçimlendiren üç temel özellik vardır. Bunlar kesim noktaları işareti, çizgi biçimi ve çizgi rengidir. `plot` fonksiyonu aksi belirtilmediği sürece kesim noktası işareti kullanmaz. Çizgi biçimi olarak düz çizgi çizgi rengi olarak ise mavi kullanır.

Çizilen bir grafiğe ait çizgi rengi, çizgi biçimi ve kesim noktası işaretini özelleştirmek için aşağıda verilen ifadeler kullanılır.

- Çizgi rengi (**Color**) seçimi yapmak için RGB kodları, renklerin İngilizce isimleri ve çok kullanılan renkler için aşağıdaki kısaltmalar kullanılabilir.

♣	Kırmızı	⇒	r	♣	Eflatun	⇒	m
♣	Yeşil	⇒	g	♣	Yeşil	⇒	y
♣	Mavi	⇒	b	♣	Siyah	⇒	k
♣	Turkuaz	⇒	c	♣	Beyaz	⇒	w

- Çizgi stili (**LineStyle**) tanımlamak için aşağıdaki kısaltmalar kullanılır.

	Kesintisiz	⇒	-
	Kesintili	⇒	-
	Noktasal	⇒	:
	Kesintili ve noktasal	⇒	-.
	Çizgisiz	⇒	none

- Kesim noktası işareti (**Marker**) seçmek için aşağıdaki kısaltmalar kullanılır.

Toplama	⇒	+	Çarpma	⇒	x
Çember	⇒	o	Üçgen (üst)	⇒	^
Nokta	⇒	.	Üçgen (alt)	⇒	v
Kare	⇒	s	Üçgen (sol)	⇒	<
Yıldız	⇒	*	Üçgen (sağ)	⇒	>
5 Köşeli Yıldız	⇒	p	6 Köşeli Yıldız	⇒	h
Elmas	⇒	d	İşaretsiz	⇒	none

Bu özelliklerle birlikte grafik çizebilmek için `plot` fonksiyonuna girilen parametreler arttırılır. Grafiğe eklenecek her bir özellik için aşağıda verilen

```
plot(x,y,'ozellik1',deger1,'ozellik2',deger2,...)
```

parametre yöntemi kullanılır.

Örnek 57 de verilen fonksiyon grafiğini yukarıdaki üç temel özelliği kullanarak yeniden çizelim. Grafiğin nasıl görüleceği Şekil 5 de figür nesnesi olmadan eksen üzerinde verilmiştir.

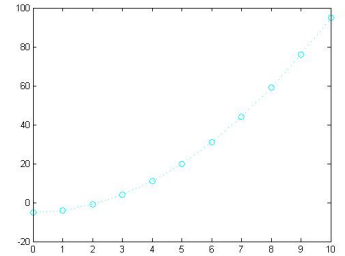
```
x = 0:1:10;  
y = x.^2 - 5;  
plot(x,y,'Color','c','LineStyle',':','Marker','o')
```

Buradaki üç özellik, bir grafiğin temel özellikleri olup, `plot` fonksiyonunda bu üç özelliği belirtmek daha kısa bir yöntem daha sunar. Bu yöntemde grafiği çizilecek vektörler verildikten hemen sonra üç temel özellik sıralamadan bağımsız olarak tek bir tırnak içerisinde

```
plot(x,y,'temel_ozellikler',...)
```

şeklinde kullanılır. Bu yöntemin kullanımında çok dikkat edilmesi gereken başka bir husus daha vardır. Grafikleri biçimlendirmek için verilecek diğer tüm özellikler (aşağıda verilen çizgi kalınlığı gibi), temel özelliklerden sonra verilmelidir. Şimdi yukarıda çizdiğimiz grafiği temel özellikler yöntemi ile tekrar çizelim.

```
x = 1:1:20;  
y = x.^2 -9*x -20;  
plot(x,y,'c:o')
```



Şekil 5: Örnek 57 de verilen y fonksiyonunun grafiği.

Matlab'da çizilen bir grafiğin çizgi kalınlığını değiştirmek için **LineWidth** özelliği ve bu özelliğe bağlı nümerik değer kullanılır. Varsayılan değer 1 birimdir. (1 birim 1/72 inc)

Örnek 58 $x = -10, x = 10$ aralığında $y = x^3 - 5x^2 + 7x + 13$ fonksiyonunun grafiğini 3 birim kalınlık, eflatun renk, 5 köşeli yıldız kesim noktası işareti ve kesintili noktasal çizgi ile çiziniz.

```
x = -10:2:10;
y = x.^3 - 5*x.^2 + 7*x + 13;
plot(x,y,'m-.p','LineWidth',3)
```

Bu örnekte x in atış miktarı 2 olarak kullanıldığından topla 11 tane kesim noktası ile grafik şekil 6 de çizilmiştir. Grafiklerde kesim noktalarını arttırmak görünümde farklılığa yol açabilir. Aynı grafik 0.1 artışla çizilirse 201 adet kesim noktası için çizilen grafik şekil 7 de verilmiştir. Kesim noktalarının fazlalığından dolayı grafik kalın bir çizgi ile çizilmiş izlenimi vermektedir.

Örnek 59 $[0, 2\pi]$ aralığında verilen x değerleri için $f(x) = \sin x$ ve $g(x) = \cos x$ fonksiyonlarının grafiklerini

- Ayrı ayrı olarak,
- Aynı eksen üzerinde birlikte,
- $f(x)$ in çizgi rengi kırmızı, çizgi stili kesintili, kesim noktası işareti +, $g(x)$ in çizgi rengi siyah, çizgi stili noktasal, kesim noktası işareti çember, çizgi kalınlıkları 2 birim olacak şekilde,
- $f(x)$ ve $g(x)$ fonksiyonlarını farklı çizgi kalınlığı ile çiziniz.

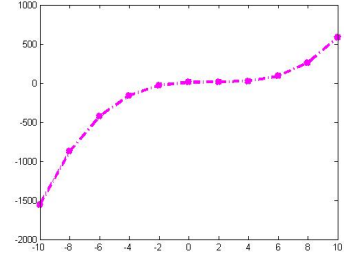
a)

```
x = 0:0.3:2*pi;
f = sin(x);
g = cos(x);
plot(x,f)
plot(x,g)
```

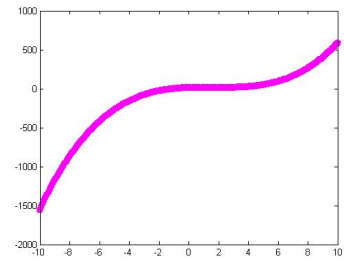
Yukarıdaki Matlab komutlarında **plot** fonksiyonu iki kez kullanılmıştır. İlk kullanımda figür nesnesi içersine oluşturulan eksende f fonksiyonunun grafiği çizilmiştir. Ancak ikinci **plot** fonksiyonu ile birlikte, şekil 8 verilen grafik üzerinde görülebileceği gibi g fonksiyonun grafiği için oluşturulan eksen f grafiğinin üzerini kapatmıştır.

b)

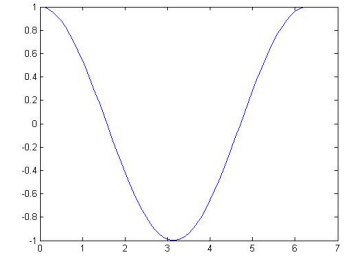
```
x = 0:0.3:2*pi;
f = sin(x);
g = cos(x);
plot(x,f,x,g)
```



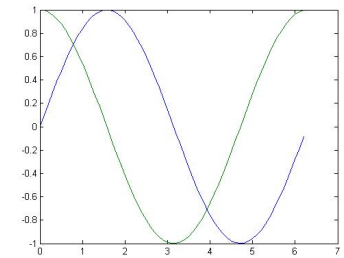
Şekil 6: Örnek 58 de verilen y fonksiyonunun grafiği.



Şekil 7: Örnek 58 de verilen y fonksiyonunun grafiği.



Şekil 8: Örnek 59-a da verilen g fonksiyonunun grafiği.



Şekil 9: Örnek 59-b de verilen f ve g fonksiyonlarının grafiği.

Yukarıdaki Matlab komutlarında bir `plot` fonksiyonu içerisinde birden fazla vektör çifti verilmiştir. Bu şekildeki kullanımda her bir vektör çiftinin aynı uzunlukta olması gerektiğine dikkat edilmelidir. Bu kurala sadık kalınarak aynı eksen üzerinde çizilen grafik sayısı artırılabilir.

```
x = 0:0.3:2*pi;
y = 1:20;
z = 10:30;
f = sin(x);
g = cos(y);
h = sin(z) + cos(z);
plot(x, f, y, g, z, h, 'LineWidth', 3)
```

Şekil 10 de görüleceği gibi, birinci grafik için mavi olan varsayılan renk, ikinci grafik için yeşil, üçüncü için ise kırmızıdır.

c)

```
x = 0:0.3:2*pi;
f = sin(x);
g = cos(x);
plot(x, f, 'r--+', x, g, 'k:o', 'LineWidth', 2)
```

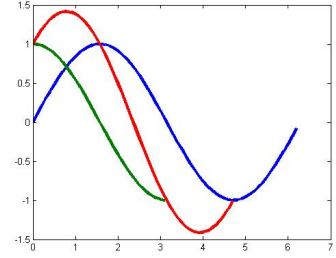
Birden fazla grafiğin aynı `plot` fonksiyonu ile çizildiği grafiklerde her bir vektör ikilisinden sonra temel özellikler tırnak içerisinde verilebilir. Ancak çizgi kalınlığı için yazılan özellik her vektör ikilisi için geçerli olacaktır.

Çizgi kalınlığını belirlemek için kullandığımız `LineWidth` gibi özellikleri temel özelliklerden önce kullanılmıyordu buna göre, aşağıdaki Matlab komutları sadece iki grafikteki çizgi kalınlıklarını 12 birim yapar. Şekil 12 da verilen grafiğe bakınız.

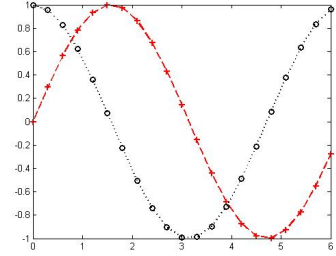
```
x = 0:0.1:2*pi;
f = sin(x);
g = cos(x);
plot(x, f, 'r--+', x, g, 'k:o', 'LineWidth', 2, 'LineWidth', 12)
```

d) Birinci grafiği 2 ikincisini ise 12 birim kalınlıkta çizmeye çalışalım. Bu işlemi yapabilmek için grafik çizimlerinde kullanılan yeni bir fonksiyona ihtiyacımız vardır. Çizilen eksen nesnesinin altta kalması, figür nesnesi üzerinde tutulması için kullanılan `hold` fonksiyonu `on` parametresi ile açılır ve `off` parametresi ile kapatılır. Parametre almadan kullanılması açık ise kapat, kapalı ise aç anlamına gelir. Bu fonksiyon yardımıyla aynı eksen nesnesi üzerindeki grafiklerimize farklı özellikler ekleyebiliriz.

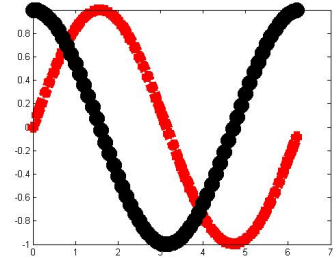
```
x = 0:0.1:2*pi;
f = sin(x);
g = cos(x);
plot(x, f, 'r--+', 'LineWidth', 2)
hold on
plot(x, g, 'k:o', 'LineWidth', 12)
```



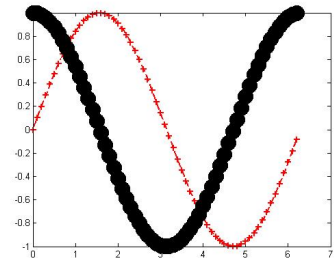
Şekil 10: Aynı eksen üzerinde çizilmiş üç farklı grafik.



Şekil 11: Örnek 59-c de verilen f ve g fonksiyonlarının grafiği.



Şekil 12: Bir plot içerisinde herhangi bir özelliğin birden fazla kullanılması son yazılan değer geçerli olması demektir.



Şekil 13: Örnek 59-d de verilen f ve g fonksiyonlarının grafiği.

`hold` fonksiyonu ile eksen nesnesini birkez figür nesnesi üzerinde tuttuktan sonra aksi belirtilmedikçe çizilecek tüm grafikler aynı eksen nesnesi üzerinde kalırlar.

Matlab'da `plot` fonksiyonu ile çizilen grafiklerde kullanılacak diğer bazı özellikler aşağıda verilmiştir.

- `MarkerSize` özelliği kesim noktası işaretçisi boyutunu belirtmek için kullanılır. Bu özelliğin varsayılan değeri 6 birimdir.
- `MarkerEdgeColor` özelliği kesim noktası işaretçisinin kenar rengini belirtmek için kullanılır. Bu özellik varsayılan olarak grafiğin çizgi rengi ile aynı değeri alır.
- `MarkerFaceColor` özelliği iç rengine sahip (dolgulu) kesim noktası işaretleri için dolgu rengini belirtmekte kullanılır. Bu özellik varsayılan olarak eksen arka planı rengi ile aynı değeri alır.

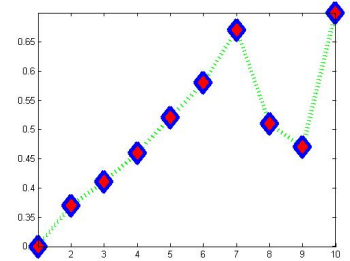
Örnek 60 *Bir x maddesi için yapılan deneylerde ölçülen değerler aşağıdaki tabloda verilmiştir.*

<i>Deney No</i>	<i>Ölçülen Değer</i>
1	0.30
2	0.37
3	0.41
4	0.46
5	0.52
6	0.58
7	0.67
8	0.51
9	0.47
10	0.70

Bu tablonun grafiğini Matlab yardımıyla çizelim.

```
degerler = [0.30,0.37,0.41,0.46,0.52,...
0.58,0.67,0.51,0.47,0.70];
plot(degerler,':gd','MarkerSize',18,...
'MarkerEdgeColor','blue',...
'MarkerFaceColor',[1.0,0.0,0.0],...
'LineWidth',5)
```

Yukarıdaki Matlab komutlarından anlaşılacağı gibi `plot` komutu ile tek bir vektörün grafiği de çizilebilir. Bu durumda vektör elemanlarını indisleri ikinci vektörmüş gibi düşünülür. Yukarıdaki Matlab komutlarıyla çizilen grafik için çizgi rengi yeşil, kesim noktası işaretleri sınır rengi mavi, dolgu rengi, RGB kodu yardımıyla kırmızı ve kesim noktası işaretleri boyutunu 18 birim olarak ayarladık. Eğer burada



Şekil 14: Örnek 60 de verilen x maddesi deneylerinin grafiği.

'+' gibi dolgunsuz bir kesim noktası işareti kullansaydık grafiğimizde kırmızı renk hiç görünmezdi.

Matlab herhangi bir vektörün grafiğini çizebildiğine göre `rand` fonksiyonu ile rastgele oluşturulan bir vektörün grafiğini de çizebilir. Şekil 15 de verilen grafiğe bakınız.

```
x = rand(1,200);
plot(x, 'cyand', 'MarkerSize', 13, ...
'MarkerEdgeColor', 'blue', ...
'MarkerFaceColor', [1.0, 0.5, 0.0])
```

`linspace(m,n)` fonksiyonu ile m ve n sayıları arasında doğrusal olarak eşit aralıklarla dağılmış 100 elemanlı bir vektör elde edebileceğimizi önceki bölümlerden biliyoruz. Şimdi bu fonksiyonu kullanarak bir karmaşık sayıda bir grafik çizelim. Aşağıdaki Matlab komutlarında `pol2cart` fonksiyonu ile polar koordinatları kartezyen koordinatlara çeviriyoruz. Şekil 16 de verilen grafiği inceleyiniz.

```
r = linspace(0,2);
t = linspace(0,10*pi);
[x,y] = pol2cart(t,r);
z = x + i*y;
plot(z)
grid on
```

Matlab'da oluşturulan eksenler üzerinde kılavuz çizgileri oluşturmak için `grid` fonksiyonu kullanılır. Kılavuz çizgilerini açmak için `grid on`, kapatmak için `grid off` parametresi kullanılır.

Örnek 61 $y = \sin\left(\frac{x}{3}\right) + \cos\left(\frac{x}{2}\right)$ fonksiyonunun grafiğini x değerleri 0.4 artış ile

a) $[-10, 10]$ aralığında, kesintili noktasal çizgi stili, 5 köşeli yıldız kesim noktası işareti, 14 birim işaretçi boyutu, çizgi rengi, işaretçi sınır rengi işaretçi dolgu rengi RGB olarak verilecek şekilde çiziniz.

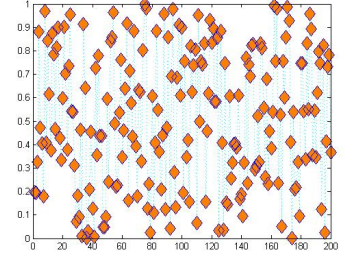
b) Fonksiyonun periyodu p ise $[-p, p]$ aralığında 0.4 artış ile çiziniz.

a) Bu örnekte klavuz çizgileri kullanılmıştır.

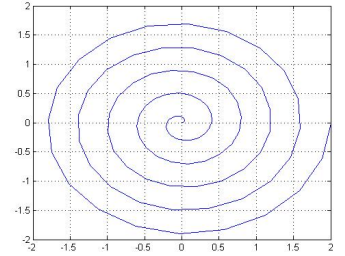
```
x = -10:0.4:10;
y = sin(x/3) + cos(x/2);
plot(x,y, '-p', 'MarkerSize', 14, 'Color', [1.0, 0.0, 0.0], ...
'MarkerEdgeColor', [0.0, 1.0, 0.0], ...
'MarkerFaceColor', [0.0, 0.0, 1.0])
grid on
```

b) İlk olarak periyodumuzu hesaplayalım. $p = \text{Okek}(6\pi, 4\pi) = 12\pi$ olduğuna göre $[-12\pi, 12\pi]$ aralığında grafiğimizi çizelim

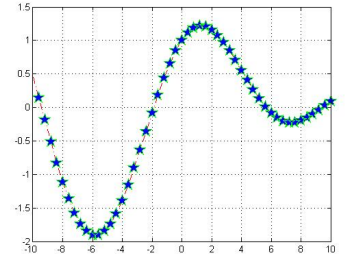
```
x = -12*pi:0.4:12*pi;
y = sin(x/3) + cos(x/2);
```



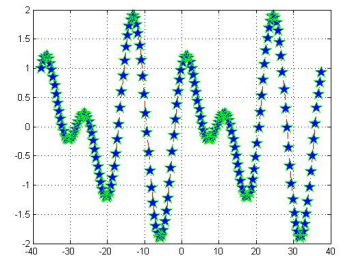
Şekil 15: Rastgele oluşturulan bir vektörün grafiği.



Şekil 16: Polar koordinatlardan kartezyen koordinatlara çevrilen vektör grafiği.



Şekil 17: Örnek 61-a da verilen y fonksiyonunun grafiği.



Şekil 18: Örnek 61-b da verilen y fonksiyonunun grafiği.

```
plot(x,y,'-p','MarkerSize',14,'Color',[1.0,0.0,0.0],...
'MarkerEdgeColor',[0.0,1.0,0.0],...
'MarkerFaceColor',[0.0,0.0,1.0])
grid on
```

Grafik Biçimlendirme Fonksiyonları

Matlab'da iki boyutlu grafikler çizilirken eksen ve grafik üzerine etki eden birçok önemli fonksiyon vardır. Bu bölümde bu özelliklerden bazılarını vereceğiz.

- `title` fonksiyonu eksene isim vermek için kullanılır.
- `xlabel` fonksiyonu x eksenine isim vermek için kullanılır.
- `ylabel` fonksiyonu y eksenine isim vermek için kullanılır.

```
title('y = sin(x/3)+cos(x/2)');
xlabel('x eksenini');
ylabel('y eksenini');
```

Şekil 19 de verilen grafiği inceleyiniz.

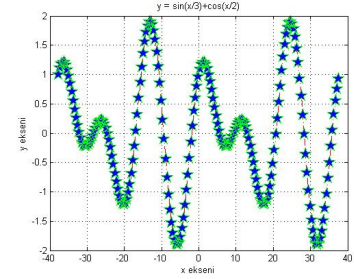
- `legend('birinci_veri','ikinci_veri',...,pos)` fonksiyonu çoklu grafiklerle çizgi özellikleri örneği ile birlikte, her bir grafik için istenilen tanımlamaları figür nesnesi üzerine yazdırır. Bu tanımlamaların figür nesnesi üzerinde nereye yapılacağına $pos = -1,0,1,2,3,4$ değerlerine göre karar verilir. Bu değerler sırası ile

- | | | |
|----|---|----------------------|
| -1 | ⇒ | Sağda grafik dışında |
| 0 | ⇒ | Grafik içinde |
| 1 | ⇒ | Sağ üstte |
| 2 | ⇒ | Sol üstte |
| 3 | ⇒ | Sol altta |
| 4 | ⇒ | Sağ altta |

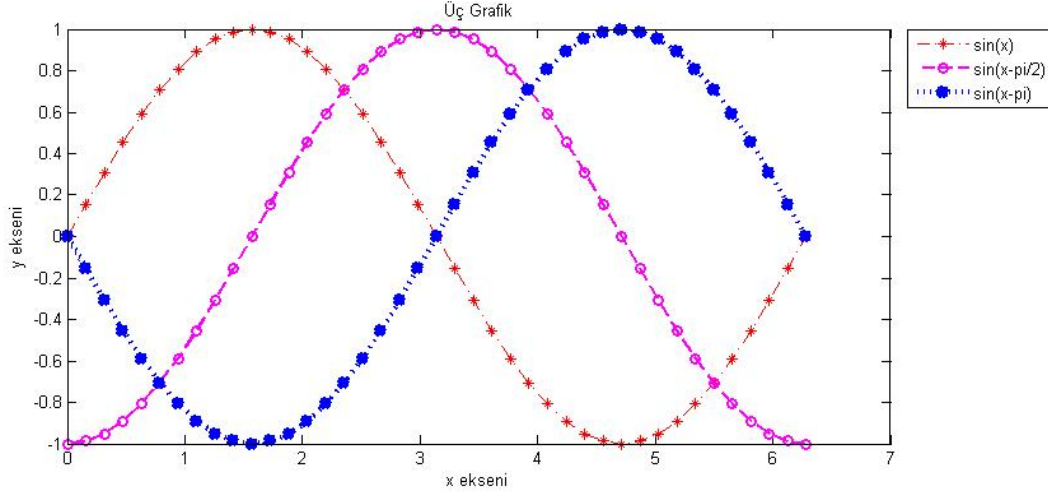
anlamlarına gelir.

Örnek 62 $[0, 2\pi]$ aralığında $\frac{\pi}{20}$ artış ile aynı eksen üzerinde $\sin(x), \sin(x - \frac{\pi}{2})$ ve $\sin(x - \pi)$ fonksiyonlarının grafiklerini değişik renk, çizgi biçimi, kesim noktası işareti ve çizgi kalınlıkları ile çizerek, herbir grafiği etikeleyiniz. Ayrıca grafik ve eksen isimlendirmeleri yapınız.

```
x = 0:pi/20:2*pi;
plot(x, sin(x), '-.r*', 'LineWidth', 1);
hold on
plot(x, sin(x-pi/2), '--mo', 'LineWidth', 2);
plot(x, sin(x-pi), ':bp', 'LineWidth', 3);
legend('sin(x)', 'sin(x-pi/2)', 'sin(x-pi)', -1)
title('Üç Grafik')
xlabel('x eksenini')
ylabel('y eksenini')
```



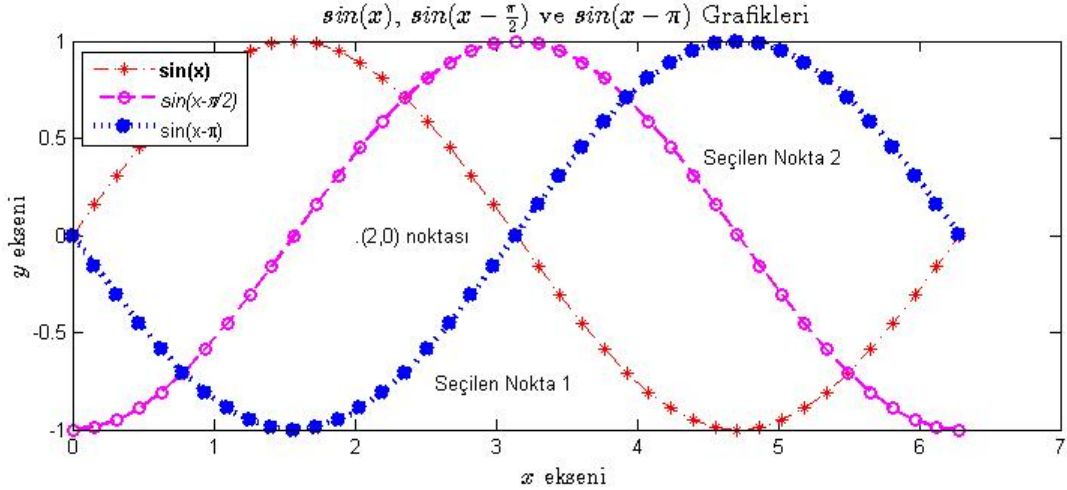
Şekil 19: Örnek 61 da verilen y fonksiyonunun isimlendirilmiş grafiği.



Şekil 20: Örnek 62 de verilen fonksiyonların grafiği.

- `text(x,y,'metin')` fonksiyonu eksen üzerinde verilen (x, y) koordinatlarına bir metin yerleştirmek için kullanılır.
- `gtext('metin')` fonksiyonu eksenin herhangi bir noktasına istenilen metni yerleştirmek için kullanılır. Grafik çizildiğinde hedef seçmek için bir yardımcı otomatik olarak başlayacaktır.
- `ginput(n)` fonksiyonu eksen üzerinde n tane noktanın koordinatları bir matrise atamak için kullanılır. Grafik çizildiğinde bu noktaları seçmek için yardımcı otomatik olarak başlayacaktır.
- Eksen üzerinde kullanılan metinsel ifadeler, önceki bölümlerde verilen Matlab'a özgü metin biçimlendirme komutları yardımıyla düzenlenebilir. Ayrıca matematiksel yazım ve tasarım dili olan LaTeX, Matlab içerisinde kullanılabilir.

```
x = 0:pi/20:2*pi;
plot(x, sin(x), '-.r*', 'LineWidth', 1);
hold on
plot(x, sin(x-pi/2), '--mo', 'LineWidth', 2);
plot(x, sin(x-pi), ':bp', 'LineWidth', 3);
legend('\bf{sin(x)}', '\it{sin(x-\pi/2)}', 'sin(x-\pi)', 2)
title('$\sin(x)$, $\sin(x-\frac{\pi}{2})$ ve $\sin(x-\pi)$ Grafikleri',
'interpreter', 'latex', 'FontSize', 13)
xlabel('$x$ eksenini', 'interpreter', 'latex', 'FontSize', 12)
ylabel('$y$ eksenini', 'interpreter', 'latex', 'FontSize', 12)
text(2, 0, '(2,0) noktası')
k = ginput(1);
text(k(1), k(2), 'Seçilen Nokta 1')
gtext('Seçilen Nokta 2');
```



Şekil 21: Örnek 62 de verilen fonksiyonların Matlab metin özellikleri ve LaTeX kullanılarak çizilmiş grafiği

Matlab kitabımızın önceki bölümlerinde `get` ve `set` fonksiyonlarının öneminden bahsetmiştik. Kısaca bu fonksiyonların herhangi bir nesnenin veya elemanın, herhangi bir özelliğinin değerini almak veya yeni değer vermek için kullanıldığını hatırlatalım. 2 boyutlu grafiklerin iki önemli nesnesi figür ve eksen nesnelere. Matlab figür nesnesini `gcf` ifadesiyle eksen nesnesini ise `gca` ifadesi ile isimlendirir. Şimdi `set` ve `get` fonksiyonlarını kullanarak figür ve eksen nesnelimizin renklerini değiştirelim.

İlk olarak figure ekranımızın arka plan rengini değiştirmek için,

```
set(gcf, 'Color', 'magenta')
```

komutu kullanılabilir. Benzer şekilde eksenin arka plan rengini değiştirmek için

```
set(gca, 'Color', 'cyan')
```

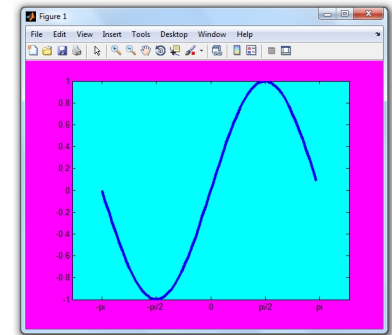
komutu kullanılabilir.

$\sin(x)$ fonksiyonunda x eksenindeki değerleri özelleştirmek için aşağıdaki Matlab kodlarını kullanabiliriz.

```
set(gcf, 'Color', 'magenta')
x = -pi:0.1:pi;
y = sin(x);
plot(x, y, 'LineWidth', 3);
set(gca, 'Color', 'cyan');
set(gca, 'XTick', -pi:pi/2:pi);
set(gca, 'XTickLabel', {'-pi', '-pi/2', '0', 'pi/2', 'pi'});
```

Alt Grafikler

Matlab'da çizilen her grafik için bir figür nesnesi ve bu nesnesin içerisinde bir eksen nesnesi oluşur. Özel durumlarda bir figür nesnesi içer-



Şekil 22: $\sin(x)$ fonksiyonunun grafiği.

sine birden fazla eksen nesnesi yerleřtirilebilir. Bu iřlemi yapabilmek iin alt grafik fonksiyonu olan `subplot` fonksiyonu kullanılır. Matlab’da gerekleřtirilen bir ok iřlemde olduėu gibi, alt grafiklerde de matrisel düşünce tarzı geerlidir. Yani fonksiyonumuz `subplot(a,b,c)` biiminde üç giriř parametresi ile alıřmaktadır. Bu parametreler

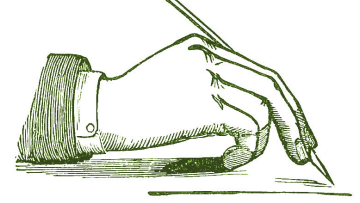
- a ⇒ Alt grafiklerin satır sayısını
- b ⇒ Alt grafiklerin sütün sayısını
- c ⇒ Alt grafiėin indis numarasını

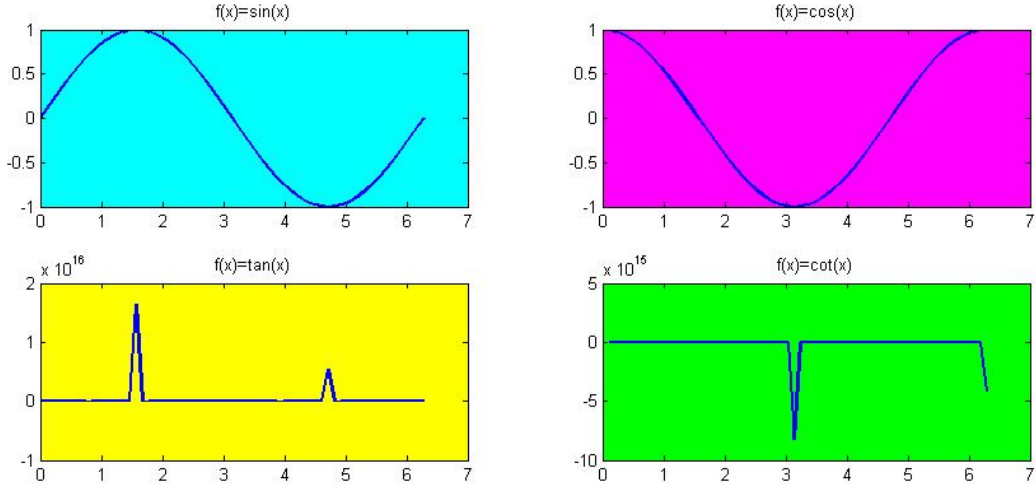
ifade eder. Birden fazla alan kaplayan alt grafikler iin indis numarası verilirken Matlab’ın liste özelliklerinden yararlanır.

Örnek 63 $[0, 2\pi]$ aralığında $\frac{\pi}{30}$ artış ile verilen x deėerleri iin aynı figür nesnesinde $\sin(x)$, $\cos(x)$, $\tan(x)$ ve $\cot(x)$ grafiklerini izen matlab komutlarını yazınız.

```
subplot(2,2,1)
x = 0:pi/30:2*pi;
y1 = sin(x);
plot(x,y1,'LineWidth',2);
title('f(x)=sin(x)');
set(gca,'Color','cyan')
subplot(2,2,2)
y2 = cos(x);
plot(x,y2,'LineWidth',2);
title('f(x)=cos(x)');
set(gca,'Color','magenta')
subplot(2,2,3)
y3 = tan(x);
plot(x,y3,'LineWidth',2);
title('f(x)=tan(x)');
set(gca,'Color','yellow')
subplot(2,2,4)
y4 = cot(x);
plot(x,y4,'LineWidth',2);
set(gca,'Color','green')
title('f(x)=cot(x)');
```

Alt grafikler oluřturulurken, matris özellikleri dikkate alınmalıdır. Örneėin ařaėıdaki Matlab komutlarıyla figür nesnesinde 4 satır ve 2 sütün eksen nesnelere iin yer ayrılmıřtır. Őekil 24 yi inceleyiniz.





Şekil 23: Alt grafik yerlesimi.

```

x = 0:0.23:2*pi;
subplot(4,2,1:2:7)
plot(sin(x), 'LineWidth', 2);
set(gca, 'Color', 'cyan')
subplot(4,2,2)
plot(cos(x), 'LineWidth', 2);
set(gca, 'Color', 'magenta')
subplot(4,2,4)
plot(tan(x), 'LineWidth', 2);
set(gca, 'Color', 'yellow')
subplot(4,2,[6,8])
plot(cot(x), 'LineWidth', 2);
set(gca, 'Color', 'green')

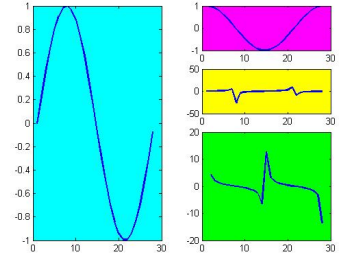
```

Aşağıdaki Matlab komutlarıyla figür nesnesinde 9 eksen nesnesi için yer ayrılmıştır. Şekil 25 ü inceleyiniz.

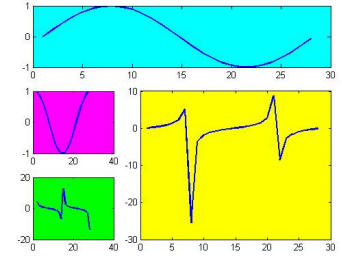
```

x = 0:0.23:2*pi;
subplot(3,3,1:3)
plot(sin(x), 'LineWidth', 2);
set(gca, 'Color', 'cyan')
subplot(3,3,4)
plot(cos(x), 'LineWidth', 2);
set(gca, 'Color', 'magenta')
subplot(3,3,[5,6,8,9])
plot(tan(x), 'LineWidth', 2);
set(gca, 'Color', 'yellow')
subplot(3,3,7)
plot(cot(x), 'LineWidth', 2);
set(gca, 'Color', 'green')

```



Şekil 24: Alt grafik yerlesimi.



Şekil 25: Alt grafik yerlesimi.

Diğer 2 Boyutlu Grafik Fonksiyonları

Matlab'da 2 boyutlu fonksiyin çizmek için başka fonksiyonlarda vardır. Bu bölümde daha farklı tarzda 2 boyutlu grafikler çizmek için kullanılan fonksiyonlardan bahsedeceğiz.

- $y = f(x)$ şeklinde verilen bir fonksiyonun grafiğini belirlenen limitler içerisinde çizdirmek için `fplot` fonksiyonu kullanılabilir. Fonksiyon ifadesi tırnak işaretleri arasında verilmelidir veya daha önceden bir değişkene aktarılmalıdır.

```
fplot('fonksiyon', limitler, 'cizgi_ozellikleri')
```

biçiminde kullanılır. Buradaki limitler fonksiyonun çizeceği bölgeyi ifade eder.

Örnek 64 $y = x^2 + 4 \sin(2x) - 1$ fonksiyonunun grafiğini $[-3, 3]$ aralığında çizelim.

```
fplot('x^2+4*sin(2*x)-1', [-3, 3], '--rs');
```

`fplot` fonksiyonunda istenirse y değerleri için de aralık verilebilir. Şekil 27 i inceleyiniz.

```
fplot('x^2+4*sin(2*x)-1', [-3, 3, -4, 2], '--rs');
```

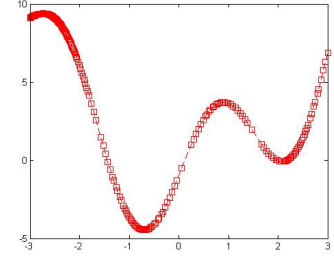
- Çizilen bir fonksiyon grafiğinde x ve y değerlerinin aralığı çok geniş olduğu durumlarda, bu değerler logaritmik artış ile tanımlanabilir. Hem x i hemde y yi logaritmik artış ile tanımlamak için `loglog`, yalnızca x veya yalnızca y yi logaritmik artış ile tanımlamak için `semilogx` veya `semilogy` fonksiyonları kullanılır.

Örnek 65 x değerleri -1000 ile 1000 aralığında 0.1 artış ile verilmek üzere

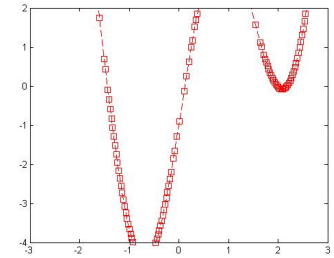
$$y = x^3 + 3x - 5$$

fonksiyonun grafiğini aynı figure nesnesinde normal olarak, x ve y logaritmik artış ile, yalnızca x logaritmik artış ile ve yalnızca y logaritmik artış ile olacak şekilde çiziniz.

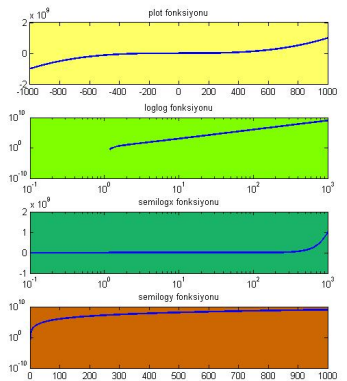
```
x = -1000:0.1:1000;
y = x.^3 + 3*x -5;
subplot(4,1,1)
plot(x,y,'LineWidth',2);
title('plot fonksiyonu')
set(gca,'Color',[1.0,1.0,0.4])
subplot(4,1,2)
loglog(x,y,'LineWidth',2);
title('loglog fonksiyonu')
```



Şekil 26: Örnek 64 da verilen y fonksiyonunun grafiği.



Şekil 27: Örnek 64 da verilen y fonksiyonunun grafiği.



Şekil 28: Örnek 65 da verilen y fonksiyonunun grafiği.

```

set(gca, 'Color', [0.5, 1.0, 0.0])
subplot(4, 1, 3)
semilogx(x, y, 'LineWidth', 2);
title('semilogx fonksiyonu')
set(gca, 'Color', [0.1, 0.7, 0.4])
subplot(4, 1, 4)
semilogy(x, y, 'LineWidth', 2);
title('semilogy fonksiyonu')
set(gca, 'Color', [0.8, 0.4, 0.0])

```

- Bazı durumlarda sayısal aralıkları farklı iki fonksiyonun grafiği aynı eksen nesnesi üzerinde çizildiğinde, birinin aldığı değerler diğerine göre çok büyük veya çok küçük olabilir. Bu durumda grafik doğru olarak yorumlanamaz. `plotyy` fonksiyonu bu gibi durumlarda ortak aralıkla birlikte kullanılır.

Örnek 66 $[0, 6\pi]$ aralığında 0.1 artış ile $f(x) = 2x^2 - 10x + 5$ ve $g(x) = \cos(x/3)$ fonksiyonlarını aynı eksende normal olarak ve `plotyy` fonksiyonu ile tek bir figür nesnesinde çiziniz.

```

x = 0:0.1:6*pi;
f = 2*x.^2-10*x+5;
g = cos(x/3);
subplot(2, 1, 1)
plot(x, f, x, g)
title('plot fonksiyonu')
subplot(2, 1, 2)
plotyy(x, f, x, g)
title('plotyy fonksiyonu')

```

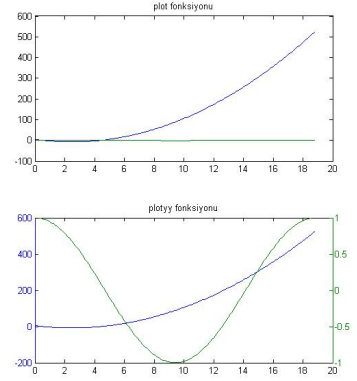
- `pie` fonksiyonu yardımıyla pasta grafikleri oluşturulabilir. Örneğin

```

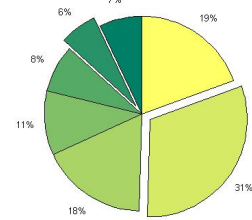
x = [8, 7, 9, 12, 20, 35, 22];
y = [0, 1, 0, 0, 0, 1, 0];
pie(x, y)
colormap summer

```

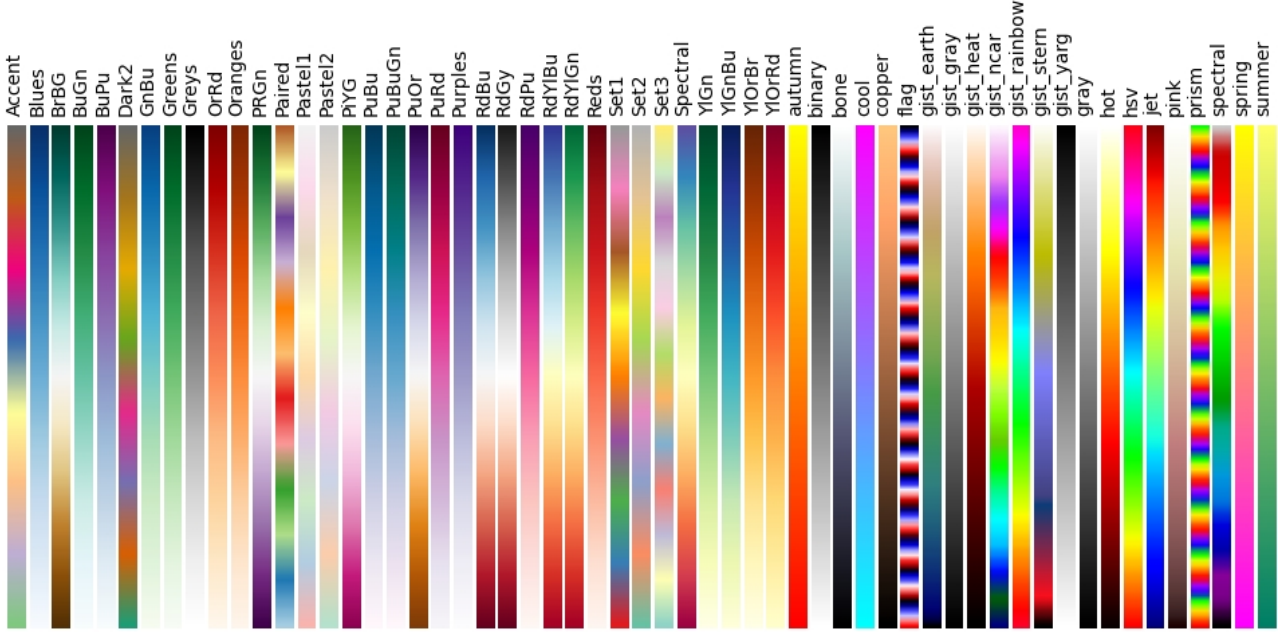
komutları pasta grafiğini çizdirir. Buradaki x değerleri grafiği çizecek veriyi, y değerleri ise pasta grafiğinde vurgu yapılacak değerleri ifade etmektedir. Biz yukarıdaki grafikte en büyük ve en küçük değere vurgu yaptık. `colormap` fonksiyonu grafikte kullanılacak renk haritasını seçmek için kullanılır. Matlab da kullanılan renk haritaları şekil 31 da verilmiştir.



Şekil 29: Örnek 66 da verilen $f(x)$ fonksiyonunun grafiği.



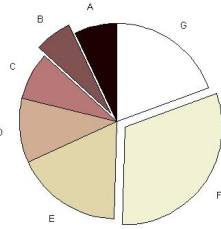
Şekil 30: Pasta Grafiği



Şekil 31: Matlab renk haritaları

`pie` fonksiyonu standart olarak her bir pasta diliminin yanında, ağırlığının yüzdesini verir. Eğer istenilirse bu alanlara başka birer metin yazılabilir.

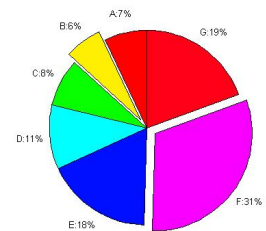
```
x = [8,7,9,12,20,35,22];
y = [0,1,0,0,0,1,0];
pie(x,y,{'A','B','C','D','E','F','G'})
colormap pink
```



Şekil 32: Pasta grafiği.

Şimdi her bir pasta dilimi için bir metin ve yüzde değerini bir arada verecek bir çözüm geliştirelim.

```
x = [8,7,9,12,20,35,22];
y = [0,1,0,0,0,1,0];
p = pie(x,y)
pyazi = findobj(p,'Type','text');
z = get(pyazi,'String')
m = {'A:','B:','C:','D:','E:','F:','G:'}
b = strcat(m,z)
set(pyazi,{'String'},b)
colormap hsv
```



Şekil 33: Pasta grafiği.

Buradaki `findobj` fonksiyonu, grafik nesnelere içerisinde belirlenen nitelikleri elde etmek için kullanılır. `strcat` fonksiyonunun ise aynı boyutlu iki metni birleştirmek için kullanıldığını hatırlayalım.

- `bar` fonksiyonu çubuk grafikleri çizmek için kullanılır. Örneğin aşağıda verilen Matlab komutları şekil 34 de verilen grafiği çizer.

```
x = [65 40 76 48 19 38 78]
bar(x)
colormap hsv
```

- `stem` fonksiyonu ile dal grafikleri ve `stairs` fonksiyonu ile basamak grafikleri çizilebilir.

Örnek 67 2008 ve 2014 yılları arasında bir A dersinden kalan öğrenci sayıları aşağıdaki tabloda verilmiştir.

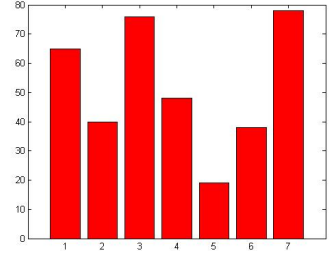
Yıllar	Kalan Öğrenci Sayısı
2008	8
2009	7
2010	9
2011	12
2012	20
2013	35
2014	22

Buna göre oluşan durumun basamak ve dal grafiklerini çiziniz.

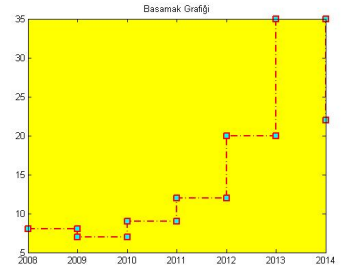
```
x = [8,7,9,12,20,35,22];
y = 2008:2014;
stairs(y,x,'-rs','MarkerFaceColor',...
'cyan','LineWidth',2);
title('Basamak Grafiği')
set(gca,'Color','yellow')
```

```
x = [8,7,9,12,20,35,22];
y = 2008:2014;
stem(y,x,'-blackd','MarkerFaceColor','cyan',...
'MarkerSize',20,'LineWidth',2);
title('Dal Grafiği')
set(gca,'Color','magenta')
```

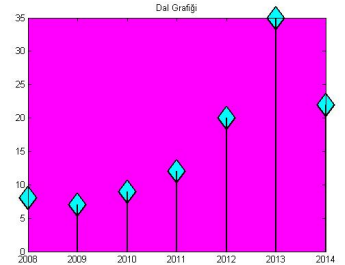
- Kutupsal koordinarlarda verilen bir fonksiyonun grafiğini çizmek için `polar` fonksiyonu kullanılır. Bu fonksiyon, t grafiğe ait noktaya karşılık gelen vektörün x eksenine yaptığı açığı ve r bu vektörün uzunluğunu belirtmek üzere `polar(t,r)` biçiminde kullanılır.



Şekil 34: Bar Grafiği

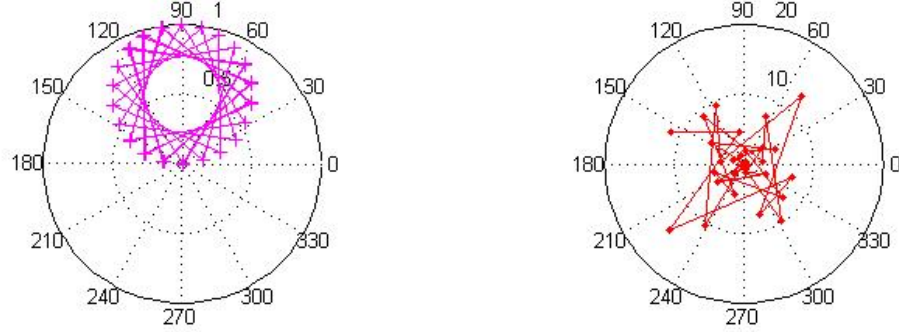


Şekil 35: Örnek 67 de verilen tablonun basamak grafiği.



Şekil 36: Örnek 67 de verilen tablonun dal grafiği.

Örnek 68 t açısı $[0, 10\pi]$ aralığında 0.1 artış ile verilmek üzere $r_1 = \sin(t)$ ve $r_2 = t \sin(t) \cos(t)$ fonksiyonlarının grafiklerini çizelim.



Şekil 37: Örnek 68 de verilen fonksiyonların grafiği.

```
t = 0:1:10*pi;
r1 = sin(t);
r2 = t.*sin(t).*cos(t);
subplot(1,2,1)
polar(t,r1,'-m+');
subplot(1,2,2)
polar(t,r2,'-r.');
```

- Metin olarak (tırnak işaretleri arasında) girilen $f(x, y) = 0$ biçimindeki kapalı fonksiyonların grafiğini çizmek için `ezplot` fonksiyonu kullanılır. Kullanım şekli

```
ezplot('kapali_fonk_ifadesi');
ezplot('kapali_fonk_ifadesi', [xmin xmax]);
ezplot('kapali_fonk_ifadesi', [xmin xmax ymin ymax]);
```

biçimindedir.

Örnek 69 $y = \frac{x^3 - 4x}{x^2 - 2x - 3}$ fonk. grafiğini `ezplot` fonksiyonu ile

a) normal olarak

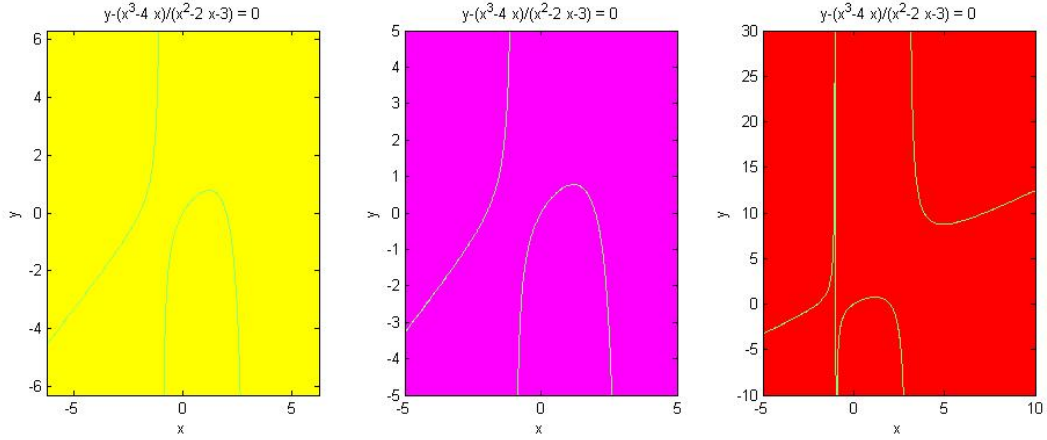
b) x değerleri $[-5, 5]$ arasında

c) x değerleri $[-5, 10]$ arasında y değerleri $[-10, 30]$ arasında çiziniz.

```
subplot(1,3,1)
ezplot('y-(x^3-4*x)/(x^2-2*x-3)')
set(gca,'Color','y')
subplot(1,3,2)
ezplot('y-(x^3-4*x)/(x^2-2*x-3)', [-5 5])
set(gca,'Color','m')
```

```
subplot(1,3,3)
ezplot('y-(x^3-4*x)/(x^2-2*x-3)', [-5 10 -10 30])
set(gca, 'Color', 'r')
```

Şekil 38 dan görülebileceği gibi ezplot ile çizilen fonksiyonların varsayılan çizgi rengi yeşildir. Ayrıca grafik ve eksen isimlendirmesi otomatik olarak yapılmaktadır. Bu özellikleriyle birlikte çok kullanılan olan **ezplot** fonksiyonu Matlab'ın sembolik hesaplama paketinin bir fonksiyonudur.



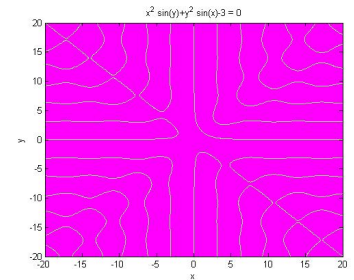
Şekil 38: Örnek 69 de verilen fonksiyonun grafikleri.

Örnek 70 $x^2 \sin y + y^2 \sin x = 3$ ifadesinin grafiğini x ve y değerlerinin her ikisinde $[-20, 20]$ aralığında çiziniz.

```
ezplot('x^2*sin(y)+y^2*sin(x)-3', [-20 20 -20 20])
set(gca, 'Color', 'm')
```

Kaynaklar

1. Stephen J. Chapman, *Matlab Programming for Engineers*, Thomson Press, 2008.
2. Stormy Attaway, *A Practical Introduction to Programming and Problem Solving*, Elsevier Press, 2012.
3. Dr. Aslan İnan, *Matlab Uygulama Çözümleri*, Papatya Yayıncılık, 2012.
4. İrfan Kaymaz, *Matlab ile Grafik Çizimi*, Ders Notları.



Şekil 39: Örnek 70 de verilen fonksiyonun grafiği.

Konu Sonu Soruları

Problem 18 $y = 1 - e^{-x} \sin(2\pi x)$ fonksiyonunun grafiğini $x \in [1, 5]$ aralığında 23 nokta ile çizen Matlab komutlarını yazınız.

Problem 19 $f(t) = 2 \sin(3t) \cos(5t)$ fonksiyonunun grafiğini $t \in [0, 3]$ aralığında 0.05 artımla çizen Matlab komutlarını yazınız.

Problem 20 $f(x) = e^{-x/10} \sin^2(x)$ fonksiyonunun grafiğini çizen bir Matlab komutlarını yazınız.

Problem 21 Matlab komutlarıyla bir daire grafiği çiziniz.

Problem 22 $y = \sin(x)$ ve $y = x^2 - 1$ fonksiyonlarını $x \in [0, 2]$ aralığında çizdirerek kesiştikleri noktayı grafik üzerinde bulan bir Matlab programı yazınız.

Problem 23 $2g(1 + \cos \theta)$ denklemini, bir kardioid mikrofonun kazançtır. Burada $g = 0.5$ mikrofon kazanç sabiti, $\theta \in [0, 2\pi]$ radyan olarak mikrofon eksenini ile ses kaynağı arasında açıdır. Mikrofon kazancının polar değişimini çizen Matlab komutlarını yazınız.

Problem 24 $y = \frac{\sin(1-x)}{1+5\cos^3 x}$ fonksiyonunun grafiğini x değerleri $[-4\pi, 4\pi]$ aralığında $\pi/50$ artış ile çizen Matlab komutlarını yazınız.

Problem 25 $y_1 = e^{-2\cos x}$ ve $y_2 = e^{1-2\sin x}$ grafiklerini üst üste $[-5, 5]$ aralığında 0.1 artışla aynı eksen üzerinde çizen Matlab komutlarını yazınız.

Problem 26 $f(x) = e^{-1/x} + \log(x^3 + 2)$ fonksiyonunun grafiğini 0 ile 20 aralığında 10 kesim noktası, 3 birim kalınlıkta kırmızı çizgi rengi, 6 köşeli yıldız kesim noktası işareti ve işaret özellikleri herhangi RGB kodları ile verilecek şekilde çizen Matlab kodlarını yazınız.

Problem 27 $[0, 1]$ aralığında 0.01 artışla verilen x değerleri için aynı grafik penceresinde $y = \sin(4\pi t)$, $z = \sin(20\pi t)$, $w = y + z$ fonksiyonlarının grafiklerini alt grafikler kullanarak ve sırasıyla 'Birinci Harmonik', 'İkinci Harmonik', 'Harmonik Dalgaların Toplamı' şeklinde isimlendirerek çizen Matlab komutlarını yazınız.

Problem 28 Avrupa, Amerika, Asya ve Afrika kıtalarında ürün satışı yapan bir firmanın satış rakamları sırasıyla 1200, 500, 300 ve 120 dir. Buna göre firmanın kıtalara göre satış oranlarını gösteren pasta grafiğini çizen Matlab komutlarını yazınız.

Problem 29 $(x^2 + y^2)^2 - 2(x^2 - y^2) = 0$ fonksiyonunun grafiğini çizen Matlab komutlarını yazınız.



Problem 30

$$x = \frac{3}{2} \cos(t) - \cos(30t)$$

$$y = \frac{3}{2} \sin(t) - \sin(30t)$$

biçiminde verilen parametrik denklemlerin grafiğini çizen Matlab komutlarını yazınız.

Çözümler

Çözüm 18 *Aşağıdaki Matlab komutlarının oluşturduğu grafik için şekil 40 i inceleyiniz.*

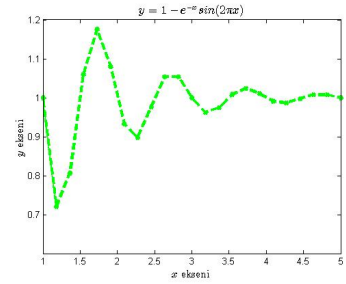
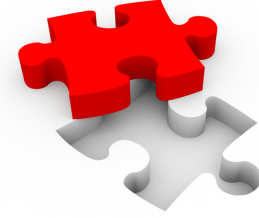
```
x=linspace(1,5,23);
y=1-exp(-x).*sin(2*pi*x);
plot(x,y,'--gx','LineWidth',3)
title('$y=1-e^{-x}\sin(2\pi x)$',...
'interpreter','latex','FontSize',13)
xlabel('$x$ ekseni','interpreter','latex','FontSize',12)
ylabel('$y$ ekseni','interpreter','latex','FontSize',12)
```

Çözüm 19 *Aşağıdaki Matlab komutlarının oluşturduğu grafik için şekil 41 u inceleyiniz.*

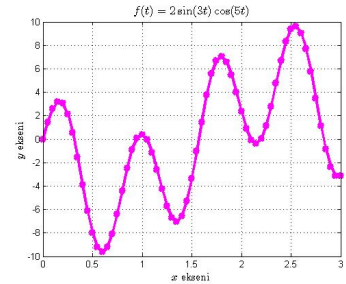
```
t=0:0.05:3;
f=10*sin(3*t).*cos(5*t);
plot(t,f,'-mp','LineWidth',3)
title('$f(t)=2\sin(3t)\cos(5t)$',...
'interpreter','latex','FontSize',13)
xlabel('$x$ ekseni','interpreter','latex','FontSize',12)
ylabel('$y$ ekseni','interpreter','latex','FontSize',12)
grid on
```

Çözüm 20 *Aşağıdaki Matlab komutlarının oluşturduğu grafik için şekil 42 i inceleyiniz.*

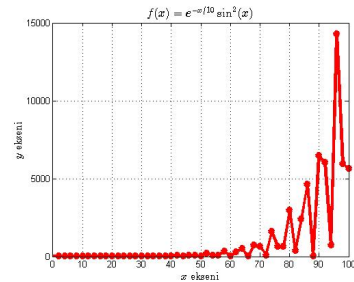
```
x=[0:2:100];
y=exp(x/10).*(sin(x)).^2;
plot(x,y,'-rh','LineWidth',3)
title('$f(x)=e^{-x/10}\sin^2(x)$',...
'interpreter','latex','FontSize',13)
xlabel('$x$ ekseni','interpreter','latex','FontSize',12)
ylabel('$y$ ekseni','interpreter','latex','FontSize',12)
grid on
```



Şekil 40: Çözüm 18 grafiği.



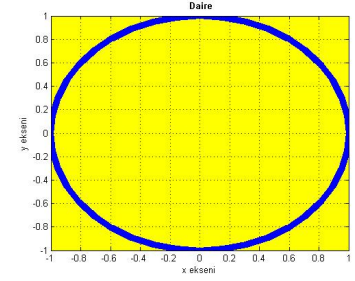
Şekil 41: Çözüm 19 grafiği.



Şekil 42: Çözüm 20 grafiği.

Çözüm 21 Aşağıdaki Matlab komutlarının oluşturduğu grafik için şekil 43 i inceleyiniz.

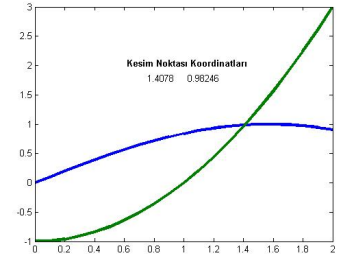
```
t=0:pi/20:2*pi;
plot(sin(t),cos(t),'LineWidth',7)
grid on
title('\bf Daire');
xlabel('x eksenini');
ylabel('y eksenini');
set(gca,'Color','y')
```



Şekil 43: Çözüm 21 grafiği.

Çözüm 22 Aşağıdaki Matlab komutlarının oluşturduğu grafik için şekil 44 yi inceleyiniz.

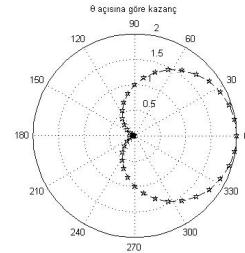
```
x=0:0.1:2;
y1=sin(x);
y2=x.^2-1;
plot(x,y1,x,y2,'LineWidth',3)
kesim_noktasi = ginput(1);
gtext('\bf Kesim Noktası Koordinatları')
gtext(num2str(kesim_noktasi))
```



Şekil 44: Çözüm 22 grafiği.

Çözüm 23 Aşağıdaki Matlab komutlarının oluşturduğu grafik için şekil 45 ü inceleyiniz.

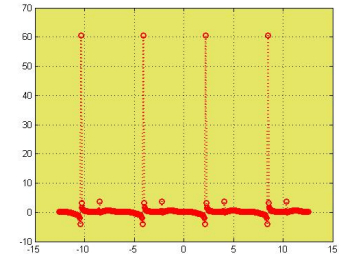
```
g=0.5;
teta=0:pi/20:2*pi;
kazanc=2*g*(1+cos(teta));
polar(teta,kazanc,'--blackp')
title('\theta açısına göre kazanç')
set(gca,'Color','r')
```



Şekil 45: Çözüm 23 grafiği.

Çözüm 24 Aşağıdaki Matlab komutlarının oluşturduğu grafik için şekil 46 ü inceleyiniz.

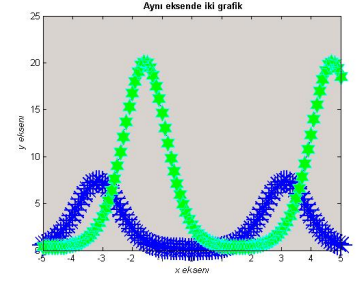
```
x=-4*pi:pi/50:4*pi;
y=sin(1-x)/(1+5*cos(x).^3);
plot(x,y,'o:r','LineWidth',2)
grid on
set(gca,'Color',[0.9,0.9,0.4])
```



Şekil 46: Çözüm 24 grafiği.

Çözüm 25 Aşağıdaki Matlab komutlarının oluşturduğu grafik için *şekil 47* i inceleyiniz.

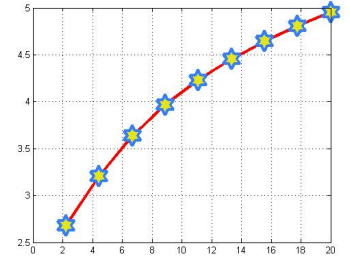
```
x=-5:0.1:5;
y1=exp(-2*cos(x));
plot(x,y1,'b*-','LineWidth',2,'MarkerSize',24)
hold on
y2=exp(1-2*sin(x));
plot(x,y2,'bh-.','LineWidth',1,...
'MarkerEdgeColor','c','MarkerSize',18,...
'MarkerFaceColor','g')
set(gca,'Color',[0.85,0.83,0.81])
title('\bf Aynı ekseninde iki grafik');
xlabel('\it x eksenini')
ylabel('\it y eksenini')
```



Şekil 47: Çözüm 25 grafiği.

Çözüm 26 Aşağıdaki Matlab komutlarının oluşturduğu grafik için *şekil 48* yi inceleyiniz.

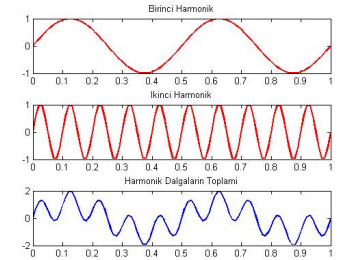
```
x=linspace(0,20,10);
y=exp(1./x)+log10(x.^3+2);
plot(x,y,'LineWidth',3,'Marker','h',....
'Markersize',23,'Color','red',....
'MarkerEdgeColor',[0.2,0.5,1.0],....
'MarkerFaceColor',[0.9,0.9,0.1])
grid on
```



Şekil 48: Çözüm 26 grafiği.

Çözüm 27 Aşağıdaki Matlab komutlarının oluşturduğu grafik için *şekil 49* yi inceleyiniz.

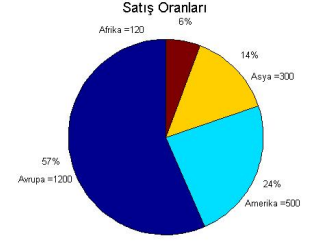
```
t=0:0.01:1;
subplot(3,1,1)
y=sin(4*pi*t);
plot(t,y,'r','LineWidth',2)
title('Birinci Harmonik')
subplot(3,1,2)
z=sin(20*pi*t)
plot(t,z,'r','LineWidth',2)
title('İkinci Harmonik')
subplot(3,1,3)
w=y+z;
plot(t,w,'r','LineWidth',2)
title('Harmonik Dalgaların Toplamı')
```



Şekil 49: Çözüm 27 grafiği.

Çözüm 28 Aşağıdaki Matlab komutlarının oluşturduğu grafik için şekil 50 i inceleyiniz.

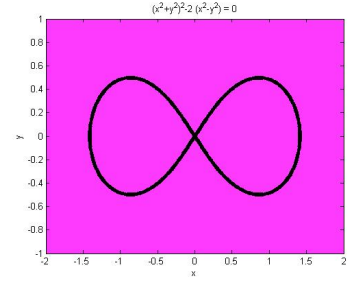
```
names = {'Avrupa = ', 'Amerika = ', 'Asya = ', 'Afrika = '};
data = [1200, 500, 300, 120];
pie(data)
for i=1:4
    gtext(strcat(names{i}, num2str(data(i))));
end
title('Satış Oranları', 'fontsize', 15)
```



Şekil 50: Çözüm 28 grafiği.

Çözüm 29 Aşağıdaki Matlab komutlarının oluşturduğu grafik için şekil 51 u inceleyiniz.

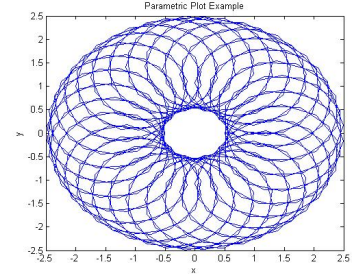
```
h = ezplot('(x^2+y^2)^2-2*(x^2-y^2)', [-2, 2, -1, 1]);
set(h, 'Color', 'black', 'LineWidth', 4);
set(gca, 'Color', [1.0, 0.23, 1.0])
```



Şekil 51: Çözüm 29 grafiği.

Çözüm 30 Aşağıdaki Matlab komutlarının oluşturduğu grafik için şekil 52 yi inceleyiniz.

```
t=linspace(0,4*pi, 600);
x = 1.5*cos(t) - cos(30*t);
y = 1.5*sin(t) - sin(30*t);
plot(x,y);
xlabel('x'), ylabel('y')
title('Parametric Plot Example')
```



Şekil 52: Çözüm 30 grafiği.