KOCAELİ ÜNİVERSİTESİ
BİLGİSAYAR MÜHENDİSLİĞİ

Bilgisayar Mühendisliğinde Matematik Uygulamaları

10. Hafta

Yrd. Doç. Dr. A. Burak İNNER

Kocaeli Üniversitesi Bilgisayar Mühendisliği
Yapay Zeka ve Benzetim Sistemleri Ar-Ge Lab.
http://yapbenzet.kocaeli.edu.tr

MATLAB UYGULAMALARI

- Belirli bir aralıkta değişim gösteren bir fonksiyonun bazı değerleri biliniyorken, bilinmeyen diğer değerleri bulmak için İnterpolasyon Yöntemi kullanılır. MATLAB'da bunun için **interp1** ve **interp2** fonksiyonları kullanılır. **interp1** fonksiyonu veri analizi ve eğri uydurma işlemlerinde tek boyutlu ara değer hesabı yapar. Kullanımı şu şekildedir.

yi = interp1(x, y, xi, method)

- Bu yazılımda parantez içine önce değişken aralığı sonra fonksiyon değeri ve fonksiyon değeri bulunmak istenen sayı ve daha sonra method yerine interpolasyon türü girilir.

- Metotlar şunlardır:

  'nearest' : yerel parçasal interpolasyon

  'linear' : linear interpolasyon

  'cubic' - bicubic interpolation

  'spline' : spline interpolasyon

- Method türü girilmez ise lineer olarak kabul edilir. Spline ve cubic türleri linear ve nearest türüne göre daha duyarlı olduğundan çoğu uygulamada tercih edilir.

Nearest Metodu

Bu metod istenilen noktaya en yakın olan veri noktasını verir

- **Örnek :** x ve y veri değerleri aşağıdaki gibi olsun. x=0.5 ve x=1.6 olduğu anda y değerleri kaç olur?

x=0.5 için:

```
x=[-3 -2 -1 0 1 2 3];
y=[9 4 1 0 1 4 9];
y2=interp1(x,y,0.5, 'nearest')
```

```
y2 =

     1
```

Ornek1.m

x=1.6 için:

```
x=[-3 -2 -1 0 1 2 3];
y=[9 4 1 0 1 4 9];
y2=interp1(x,y,1.6, 'nearest')
```

```
y2 =

     4
```

Ornek2.m

Linear Metodu

Bu metod istenilen x değerinin solundaki ve sağındaki veri noktalarını alıp, bu noktalardan bir doğru geçirir.Bu doğrunun istenilen x değerine karşılık gelen y değerini verir. Linear interpolasyon metodu sadece birinci derece bir polinoma uyan very noktaları için en uygun sonucu vermektedir.

```
x=[-3 -2 -1 0 1 2 3];
y=[9 4 1 0 1 4 9];
y2=interp1(x,y,[0.5 1.6],'linear')
```

```
y2 =

    0.5000    2.8000
```

Ornek3.m

Cubic Metodu

Bu metod her iki veri noktasından geçen 3. derece bir polinom bulur ve bu polinomu kullanarak istenilen noktadaki y değerini hesaplar.

```
x=[-3 -2 -1 0 1 2 3];
y=[9 4 1 0 1 4 9];
y2=interp1(x,y,[0.5 1.6],'cubic')
```

```
y2 =

    0.3125    2.5480
```

Ornek4.m

Spline Metodu

```
x=[-3 -2 -1 0 1 2 3];
y=[9 4 1 0 1 4 9];
y2=interp1(x,y,[0.5 1.6],'spline')
```

```
y2 =

    0.2500    2.5600
```

Ornek5.m

Spline Metodu

- interp2 fonksiyonu iki boyutlu ara değer bulma işlemini gerçekleştirir. Bu işlem görüntü işleme ve veri görüntülenmesinde kullanılır. Kullanımı şu şekildedir:

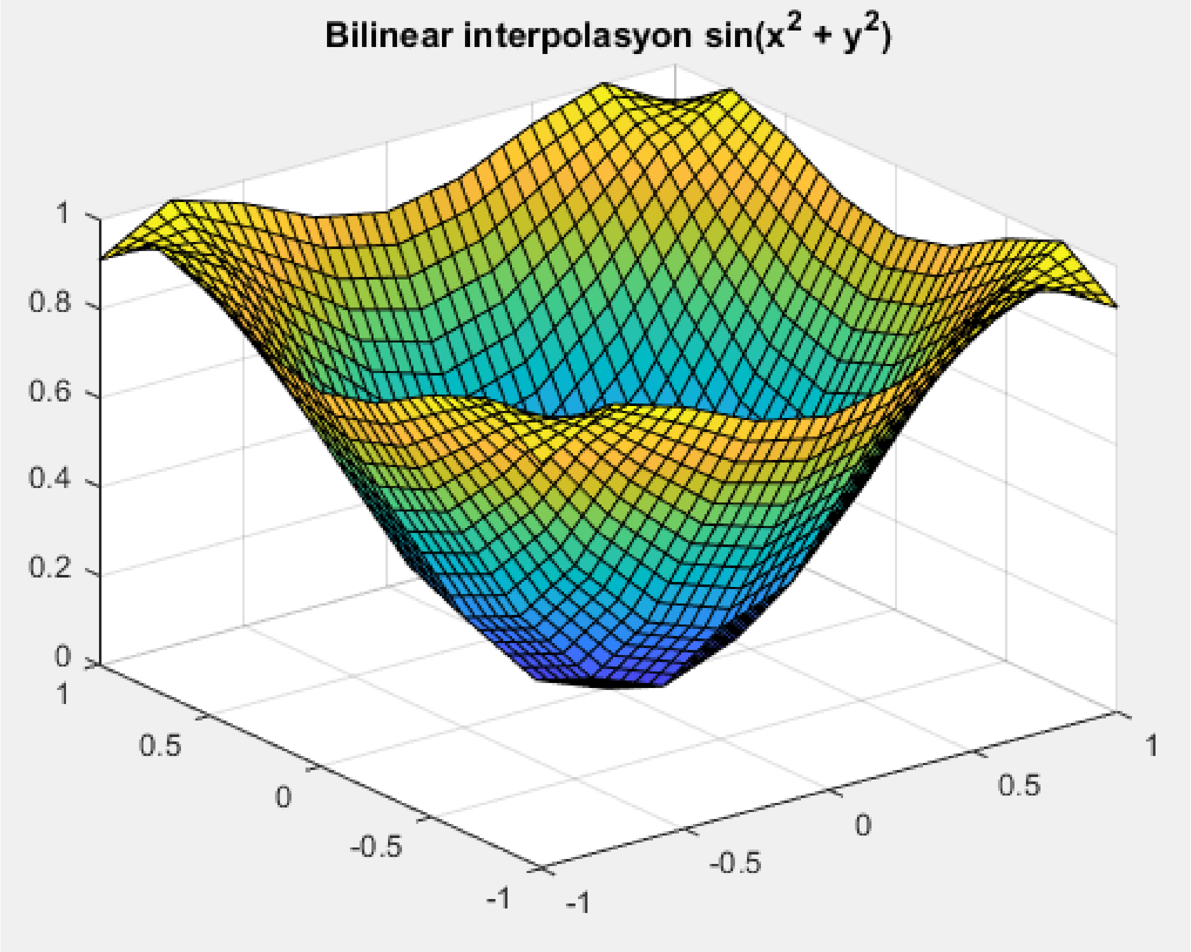
  zi = interp2(x, y, z, xi, yi, 'method’)

- Metodlar interp1 fonksiyonu ile aynı şekilde kullanılır.

Spline Metodu

- **Örnek:** z = sin(x2 + y2) fonksiyonundan –1≤ x ≤ 1, -1 ≤ y ≤ 1 aralığında data üreterek 'linear' vethe 'cubic' metotları ile interpolasyon yapalım.

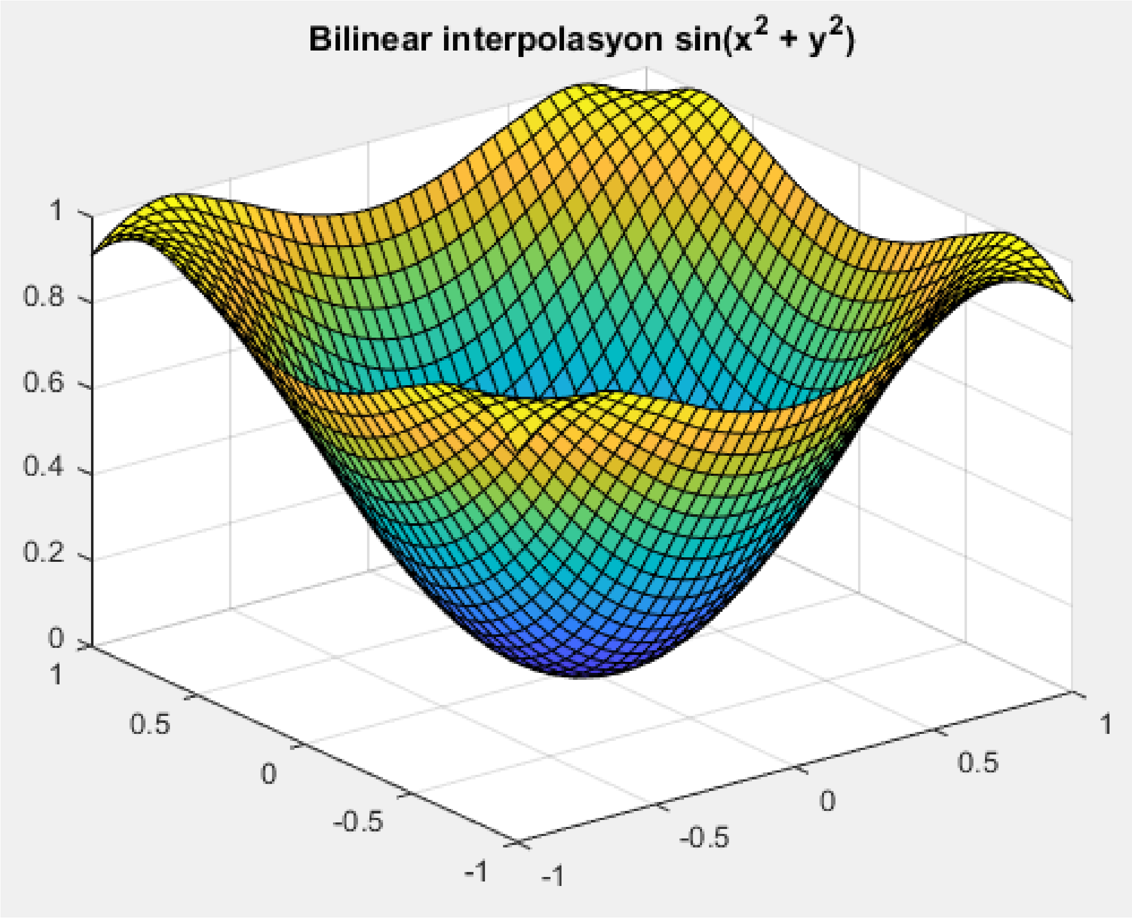
```
[x, y] = meshgrid(-1:.25:1);  z = sin(x.^2 + y.^2);
[xi, yi] = meshgrid(-1:.05:1);
zi = interp2(x, y, z, xi, yi, 'linear');
surf(xi, yi, zi), title('Bilinear interpolasyon
sin(x^2 + y^2)')
```


Bilinear interpolasyon sin(x^2 + y^2)
1
0.8
0.6
0.4
0.2
0
1
0.5
0
-0.5
-1
-1
-0.5
0
0.5
1

Ornek6.m

- cubic interpolasyon için,

```
[x, y] = meshgrid(-1:.25:1);
z = sin(x.^2 + y.^2);
[xi, yi] = meshgrid(-1:.05:1);
zi = interp2(x, y, z, xi, yi, 'cubic');
surf(xi, yi, zi), title('Bilinear interpolasyon sin(x^2 + y^2)')
```



Bilinear interpolasyon $\sin(x^2 + y^2)$
1
0.8
0.6
0.4
0.2
0
1
0.5
0
-0.5
-1
-1
-0.5
0
0.5
1

[Ornek7.m]

- Örnek : Aşağıda verilen tabloda bir motorun belli devirlerinde ve belli zamanlarda silindir başı sıcaklık değerleri gösterilmiştir. Buna göre 3.25 saniyede 3500 d/d karşılık gelen sıcaklık değerini hesaplanması.

Motor Hızıd/d	2000	3000	4000	5000
Zaman(s)	Sıcaklık(Derece)			
0	0	0	0	0
1	20	110	176	190
2	60	180	220	285
3	68	240	349	380
4	77	310	450	510

Metod belirtilmediği için linear interpolasyona göre hesaplama yapıldı.

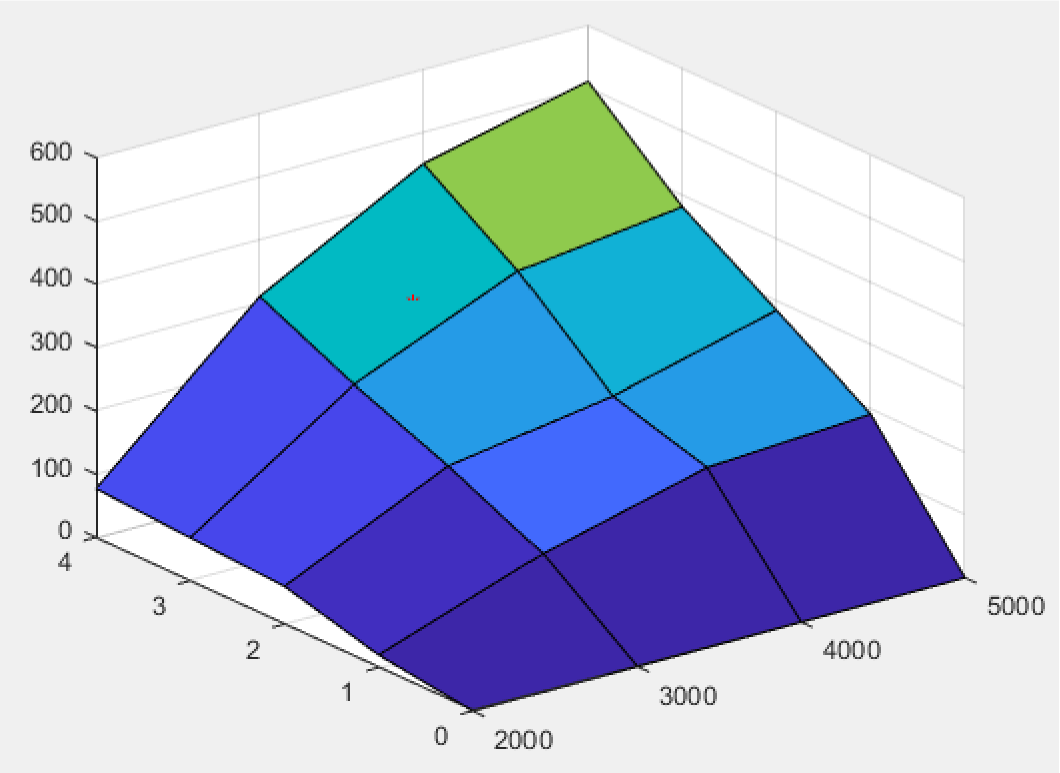
```
x=[2000 3000 4000 5000];
y=[0 1 2 3 4];
z=[0 0 0 0;
20 110 176 190;
60 180 220 285;
68 240 349 380;
77 310 450 510];
sicaklik=interp2(x,y,z,3500,3.25)
```

```
sicaklik =

  315.8750
```

Ornek8.m

Bulunan sicaklik değerinin 3B yüzey üzerinde nereye karşılık geldiği aşağıdaki şekilden görülebilir. Hesaplanan değer kırmızı * ile temsil edilmektedir.

```
x=[2000 3000 4000 5000];
y=[0 1 2 3 4];
z=[0 0 0 0;
20 110 176 190;
60 180 220 285;
68 240 349 380;
77 310 450 510];
sicaklik=interp2(x,y,z,3500,3.25);
surf(x,y,z);hold on;
plot3(3500,3.25,sicaklik,'r*')
```



Ornek9.m

Cubic interpolasyon yapılmış olsaydı:

```
x=[2000 3000 4000 5000];
y=[0 1 2 3 4];
z=[0 0 0 0;
20 110 176 190;
60 180 220 285;
68 240 349 380;
77 310 450 510];
sicaklik=interp2(x,y,z,3500,3.25, 'cubic')
```

```
sicaklik =

  326.3477
```

Ornek10.m

Spline yöntemine göre:

```
x=[2000 3000 4000 5000];
y=[0 1 2 3 4];
z=[0 0 0 0;
20 110 176 190;
60 180 220 285;
68 240 349 380;
77 310 450 510];
sicaklik=interp2(x,y,z,3500,3.25,'spline')
```

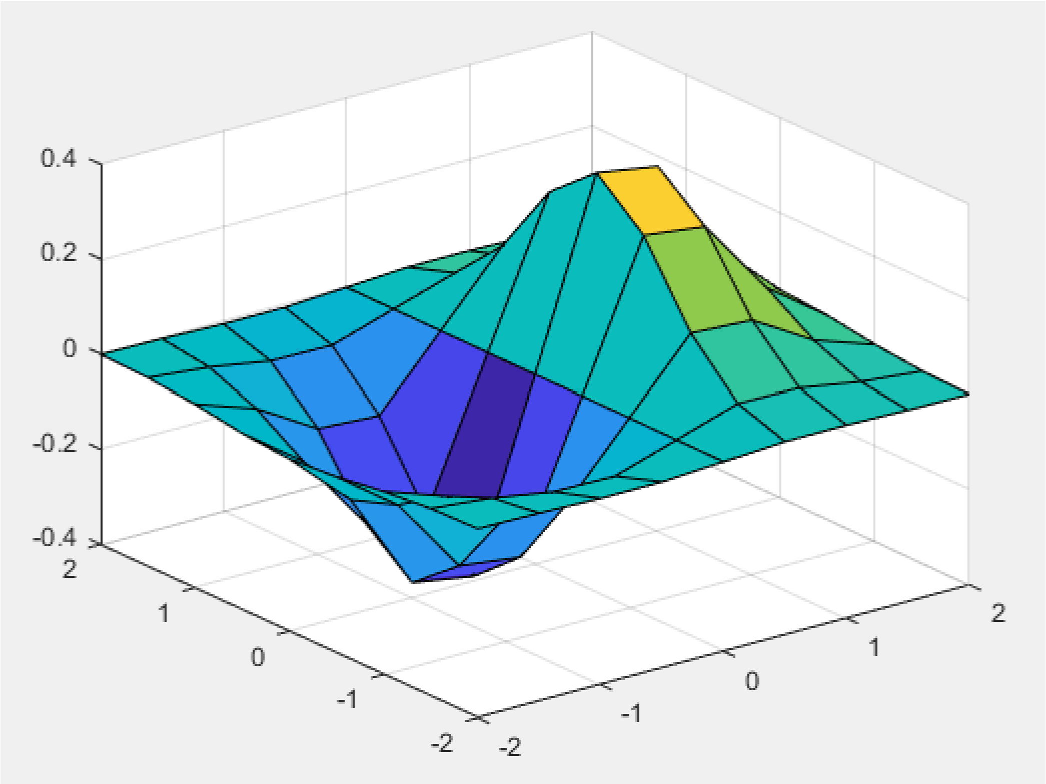
```
sicaklik =

  330.4401
```

Ornek11.m

Örnek: f(x,y)=xexp(-x2-y2) fonksiyonunun X=[-2:0.5:2] ve Y=[-2:0.5:2] aralığındaki değerlerini Z hesaplatınız ve grafiğini çizdiriniz. X, Y ve Z değerlerini kullanarak yeni X2=[-2:0.1:2] ve Y2=[-2:0.1:2] aralığında Z2 değerlerini interpolasyon yaparak hesaplatınız ve grafiğiniz çizdiriniz.

```
[X,Y]=meshgrid(-2:0.5:2);
Z=X.*exp(-X.^2-Y.^2);
surf(X,Y,Z);
```

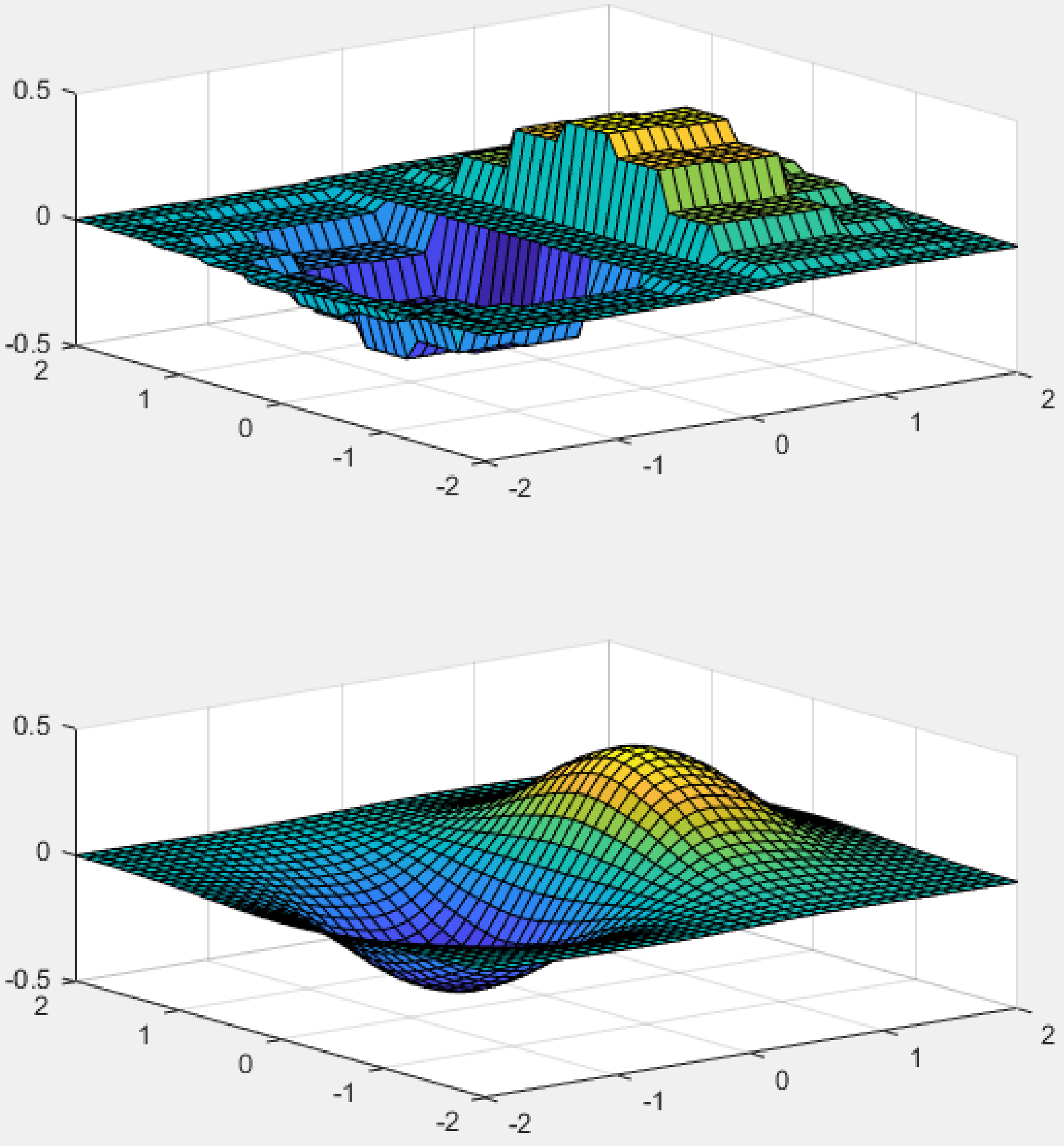
Ornek12.m



X2 ve Y2 lere göre X,Y ve Z değerleri kullanılarak Z2 değerleri farklı metodlara göre bulup çizdiren program şu şekilde yazılabilir.

```
[X,Y]=meshgrid(-2:0.5:2);
Z=X.*exp(-X.^2-Y.^2);
[X2,Y2]=meshgrid(-2:0.1:2);
Z2=interp2(X,Y,Z,X2,Y2,'nearest');
subplot(2,2,1);surf(X2,Y2,Z2);
Z2=interp2(X,Y,Z,X2,Y2,'linear');
subplot(2,2,2);surf(X2,Y2,Z2);
Z2=interp2(X,Y,Z,X2,Y2,'cubic');
subplot(2,2,3);surf(X2,Y2,Z2);
Z2=interp2(X,Y,Z,X2,Y2,'spline');
subplot(2,2,4);surf(X2,Y2,Z2);
```

Ornek13.m


0.5
0
-0.5
2
1
0
-1
-2
-2
-1
0
1
2

Örnek

- Tablodaki veriler için Newton ileri doğru interpolasyon yöntemiyle,
- A) İkinci dereceden,
- B) Üçüncü dereceden bir polinom bulunuz.
- C) Aynı yöntemle f(0.34)'ün değerini hesaplayınız.
- D) Üçüncü dereceden interpolasyon polinomu için f(0.34)'ün değerini hesaplayacak bir MATLAB programı geliştiriniz.

x_i	0.1	0.2	0.3	0.4	0.5	0.6
y_i	1.021	1.088	1.207	1.384	1.625	1.936

ÇÖZÜM

- Soruda verilen tablo verileri f(x) = x^3+ $2x^2$ + 1 fonksiyonuna yani kübik bir fonksiyona a
- Ttir. 0.34 noktasında opolinomun gerçek değeri aşağıda gösterildiği gibi 1.271’dir.

f(x) = x^3+ $2x^2$ + 1, f(0.34) = 0.34^3+ 2 x 0.34^2 + 1 = 1.271

A) Tablodaki veriler için fark tablosu 1., 2., 3. ve 4. derece fark değerleri için hazırlanır. X değişkenleri arasındaki fark, 0.1’e eşittir. Yani adım büyüklüğü; h = x_{i+1} - x_i = 0.1’dir. İleri doğru farklar, fonksiyonun değerleri temel alınarak hesaplanır:

Δf için : (1.088-1.021) = 0.067, (1.207-1.088) = 1.119 …

Δ^2f için : (0.119-0.067) = 0.052, (0.177-0.119) = 0.058 …

Δ^3f için : (0.058-0.052) = 0.006, (0.064-0.058) = 0.006 …

Δ^4f için : (0.119-0.067) = 0.052, …

x_i	f_i	Δf	Δ^2f	Δ^3f	Δ^4f
0.1	1.021	0.067	0.052	0.006	0.000
0.2	1.088	0.119	0.058	0.006	0.000
0.3	1.207	0.177	0.064	0.006	
0.4	1.384	0.241	0.070		
0.5	1.625	0.311			
0.6	1.936				

0.34 { (0.3 – 0.4)

Hazırlanan tabloda görüldüğü gibi dördüncü dereceden ileri doğru fark değerleri yani Δ^4f değerleri sıfır çıkmıştır. İkinci dereceden bir polinom için Δ f ve Δ^2f değerlerinin kullanılması yeterlidir. Böylece Newton'un ileri doğru interpolasyon yöntemi için yazılan eşitlik, $f(x) = f_0 + \alpha \Delta f_0 + \frac{\alpha(\alpha-1)}{2!} \Delta^2 f_0$, sadece 1. ve2. dereceden ileri doğru farkların kullanılmasıyla, 2. dereceden polinomun bulunması için kullanılabilir.

$$f(x) = 1.021 + \alpha\, 0.067 + \frac{\alpha(\alpha-1)}{2} 0.052 \quad \rightarrow \quad f(x) = 0.026\alpha^2 + 0.041\,\alpha + 1.021$$

B) 3. Dereceden interpolasyon için hazırlansın tablonun üçüncü sütunundaki $\Delta^3 f$ fark değerinin de kullanılması gerekir. Bu verilere göre dördüncü derece bir polinomun bulunması mümkün değildir.

$$f(x) = 1.021 + \alpha\, 0.067 + \frac{\alpha(\alpha-1)}{2}\, 0.052 + \frac{(\alpha-1)(\alpha-2)}{6}\, 0.006$$

$$f(x) = 0.001\alpha^3 + 0.023\alpha^2 + 0.043\,\alpha + 1.021$$

C) x_0'ın istenen noktaya en yakın değer olarak seçilmesi gerektiğinden x=0.34 olarak verildiğinden tabloya göre x_0=0.3 olarak seçilir. h=0.1 için α'nın değeri 0.4'dür.

$$x = x_0 + \alpha h \quad \rightarrow \quad \alpha = \frac{x - x_0}{h},$$

$$\alpha = \frac{0.34 - 0.3}{0.1} = 0.4$$

- Bu değer kullanılarak polinomu aldığı değerler ikinci ve üçüncü dereceden polinomlar için sırasıyla aşağıdaki gibi bulunur. İlave edilen fark değerleri sonucun daha hassas olarak bulunmasını sağlayacaktır.

$$f(0.34) = 1.207 + 0.4x0.177 + \frac{0.4(0.4-1)}{2}0.064 = 1.270$$

$$f(0.34) = 1.207 + 0.4x0.177 + \frac{0.4(0.4-1)}{2}0.064 + \frac{0.4(0.4-1)(0.4-2)}{6}0.006 = 1.271$$

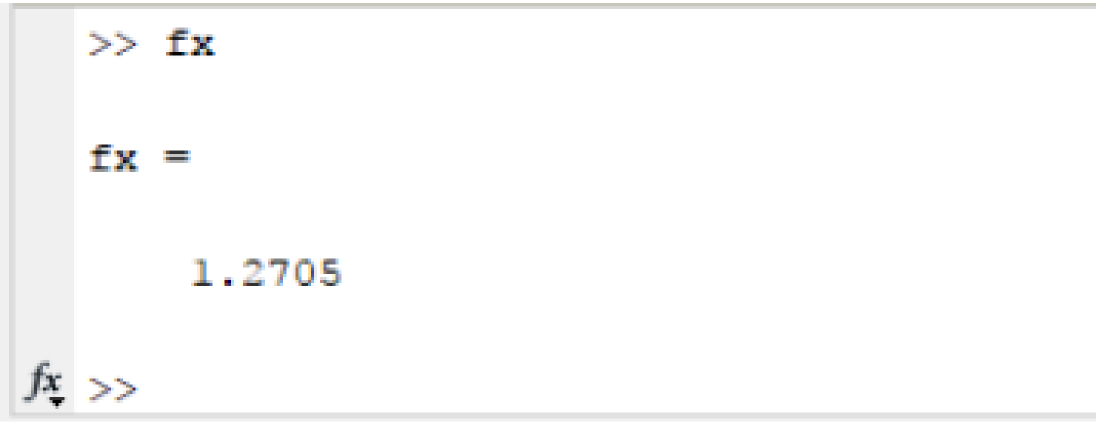
Gerçek bağıl hata, ikinci ve üçüncü dereceden polinom için sırasıyla,

$$\varepsilon_b(0.34) = \frac{1.271 - 1.270}{1.271} \cong \%0.08, \qquad \varepsilon_b(0.34) = \frac{1.271 - 1.271}{1.271} \cong \%0.0$$

Olarak hesaplanır. Beklendiği gibi gerçek polinom kübik formda olduğundan, üçüncü dereceden bulunan polinomla hesaplanan sonuç daha doğru çıkmıştır. Dördüncü dereceden fark değerleri sıfır olduğundan verilen veriler için sadece birinci, ikinci ve üçüncü dereceden polinom hesaplanabilir.

D)

```
x=[0.1 0.2 0.3 0.4 0.5 0.6];
y=[1.021 1.088 1.207 1.384 1.625 1.936];
x0=0.34;
derece=3;
n=size(x,2);
f=zeros(n,n);
for i=1: n-1
 f(i,1)=(y(i+1)-y(i));
end;
for j=2:n-1
for i=1:n-j
f(i,j)=(f(i+1,j-1)-f(i,j-1));
end;
end;
d=abs(x-x0);
ind=find(d==min(d));
farkdegerleri=[y(ind) f(ind,:)];
alfa=(x0-x(ind))/(x(2)-x(1));
fx=farkdegerleri(1)+alfa*farkdegerleri(2);
for i=1:derece
carpim=alfa;
for j=1:i
carpim=carpim*(alfa-j);
end;
fx=fx+carpim*farkdegerleri(i+2)/factorial(i+1);
end
```



[Son örnek.m]

Teşekkürler.

Dersin Sonu

Kocaeli Üniversitesi Bilgisayar Mühendisliği
Yapay Zeka ve Benzetim Sistemleri Ar-Ge Lab.
http://yapbenzet.kocaeli.edu.tr/