뛉䷴瓅

>> d=[-3 5 6];

>> D=diag(d)

D =

Diagonal Matrix

-3 0 0

0 5 0

0 0 6

>> F[0 1 8; 3 -2 -4; 0 9 6];

parse error:

syntax error

>>> F[0 1 8; 3 -2 -4; 0 9 6];

^

>> F=[0 1 8; 3 -2 -4; 0 9 6];

>> diag(F)

ans =

0

-2

6

>> A(2,:)

error: 'A' undefined near line 1 column 1

>> F(2,:)

ans =

3 -2 -4

>> F(:1)

parse error:

syntax error

>>> F(:1)

^

>> F(:,1)

ans =

0

3

0

>> i=[1,3,5];

>> j=[2,3,4];

>> v=[10 11 14];

>> S=sparse(i,j,v)

S =

Compressed Column Sparse (rows = 5, cols = 4, nnz = 3 [15%])

(1, 2) -> 10

(3, 3) -> 11

(5, 4) -> 14

>> full(S)

ans =

0 10 0 0

0 0 0 0

0 0 11 0

0 0 0 0

0 0 0 14

>> A=[4 -2 1; 3 -1 4; 1 -1 3];

>> b=[15;8;13];

>> [x,U]=gauss(A,b)

x =

-3

-13

1

U =

4.00000 -2.00000 1.00000

0.00000 0.50000 3.25000

0.00000 0.00000 6.00000

>> b=[1 3 5;]

b =

1 3 5

>> b=b'

b =

1

3

5

>> x=A/b

error: operator /: nonconformant arguments (op1 is 3x3, op2 is 3x1)

>> x=A\b

x =

-1.5833

-3.0833

1.1667

>> A=hilb(3)

A =

1.00000 0.50000 0.33333

0.50000 0.33333 0.25000

0.33333 0.25000 0.20000

>> [L,U,P]=lu(A)

L =

1.00000 0.00000 0.00000

0.33333 1.00000 0.00000

0.50000 1.00000 1.00000

U =

1.00000 0.50000 0.33333

0.00000 0.08333 0.08889

0.00000 0.00000 -0.00556

P =

Permutation Matrix

1 0 0

0 0 1

0 1 0

>> y=L\(P\*b)

y =

1.0000

4.6667

-2.1667

>>>> a=[1 -1]

a =

1. -1

੿>> b=[1 0 1;]

b =

1 0 1

>> conv(a,b)

ans =

1 -1 1 -1

A=[0 1 1; 2 -1 1; 1 1 -1];

>> x=[-2 -1 1 2];

>> y=[-6 0 0 6];

>> polyfit(x,y,3) -(katsayıları 3.dereceden bi polinom haline getir)

ans =

1.0000e+00 -3.3003e-16 -1.0000e+00 2.1579e-16

>>

\*\*\*\*\*

poly(5) = x-5

polyder

Polyval(fonksiyon,aralık) –(fonksiyonun bu aralıktaki değerlerini bul)

\*\*\*

>> plot(z,d,'o')

>> plot(z,d,'o',x,y)

>> x=[-2 -1 1 2];

>> y=[-6 0 0 6];

>> [l,L]=lagranp(x,y)

l =

1 0 -1 0

L =

-0.083333 0.166667 0.083333 -0.166667

0.166667 -0.166667 -0.666667 0.666667

-0.166667 -0.166667 0.666667 0.666667

0.083333 0.166667 -0.083333 -0.166667

>> xx=[-2:0.02:2];

>> yy=polyval(l,xx);

>> plot(xx,yy,'b',x,y,'g')

>>x=[-1 -1 1 2 4];

>> y=[0 6 60 0 -6];

>> [n,DD]=newtonp(x,y);

>> [n,DD]=newtonp(x,y)

n =

0 1 0 -1 0

DD =

0 6 7 1 -0

6 27 5 1 0

60 12 1 0 0

0 6 0 0 0

-6 0 0 0 0

>> xx=[-2:0.02:2];

>> yy=polyval(l,xx);

>> plot(xx,yy,'b',x,y,'g')

SORU

>> x=[0.25 0.75 1.25 1.5 2.0]

x =

0.25000 0.75000 1.25000 1.50000 2.00000

>> y=[0.25 0.75 0.90 1.88 4.4]

y =

0.25000 0.75000 0.90000 1.88000 4.40000

>> polyfit(x,y,4)

ans =

-4.0503 19.6099 -30.1400 17.7384 -2.5914

>> a=polyfit(x,y,4)

a =

-4.0503 19.6099 -30.1400 17.7384 -2.5914

>> polyval(a,1.80);

>> v=polyval(a,1.80);

>> plot(x,y,'b--',1.80,v);

>>