LAB 5

Interpolation

Objective

The aim of this introductory lab is to introduce you to the basic functions in the Matlab and Numerical Methods with Matlab toolbox. By the end of today's lab, you should be able to understand the Lagrange Interpolation.

Submission Requirements

You are expected to complete the assigned tasks within the lab session and show them to the lab engineer/instructor. Some of these tasks are for practice purposes only while others (marked as '*Exercise*' or '*Question*') have to be answered in the form of a lab report that you need to prepare. Following guidelines will be helpful to you in carrying out the tasks and preparing the lab report.

Guidelines

- In the exercises, you have to put the output in your Lab report. You may add screen print to the report by using the 'Print Screen' command on your keyboard to get a snapshot of the displayed output. This point will become clear to you once you actually carry out the assigned tasks**.**

- Name your reports using the following convention:

  Lab#_Rank_YourFullName

  - '#' replaces the lab number
  - '*Rank*' replaces Maj/Capt/TC/NC/PC
  - '*YourFullName*' replaces your complete name.
- You need to submit the report even if you have demonstrated the exercises to the lab engineer/instructor or shown them the lab report during the lab session.

Polynomials

Matlab stores polynomials by storing their coefficients in a row vector (3 x 1 matrix)
e.g. x^2+2x+3 would be stored as the vector [1 2 3]

>>P=[1 2 3]

Polynomials manipulation Function

The **polyval** function is used to evaluate polynomials at a particular point e.g.

Evaluates the polynomial P at the value x=2

>>polyval(P,2)
ans= 11

The **conv** function is used to multiply two polynomials e.g.
To multiply the polynomial P with (x+1) i.e. (x2+2x+3)(x+1)

>>Q=[1 1]

>>conv(P,Q)
ans= 1 3 5 3 **i.e. x^3+3x^2+5x+3**

The **poly2sym** function is used to converts a vector of coefficients to a symbolic polynomial e.g.

To convert polynomial P into symbolic form

>>syms x

>>poly2sym(P,x)
ans= x^2+2*x+3

The **polyfit** is used to find the polynomial of degree n that passes through n+1 data point e.g.

To convert polynomial P into symbolic form

```
>>x = [0 1 2 3];
>>y = sin(x*pi/6);
>>p=polyfit(x,y,3);
>>plot(x,y,'o')
>>hold on
>>syms t
>>SP=poly2sym(p,t)
>>ezplot(SP, [-1 4])
```

See the graph and analyze it.

Poly function used to construct the Lagrange Coefficients for Lagrange Polynomial, which constructs a polynomial with given roots e.g.

Construct a polynomial with roots 1 and 2

```
>>poly([1 2])
ans= 1 -3 2
```

Polynomial (x-1)(x-2) = x2-3x+2 which has roots 1 and 2

Example of Lagrange interpolating polynomial

$f(x)=x+2/x$ at point $x_0=1$, $x_1=2$, and $x_3=2.5$

Formula for the Lagrange coefficient polynomial

$y_o * L_{2,0} + y_1 * L_{2,1} + y_2 * L_{2,2}$

Where $L_{2,0} = (x-x_1)* (x-x_2)/ (x_0-x_1)* (x_0-x_2)$ and $Y=f(x)$

$L_{2,0} = (x-2)*(x-2.5)/(1-2)*(1-2.5)$

Y=[3 3 3.3]

Lagrange Coefficient polynomial in 3x3 matrix L

- 1st row is $L_{2,0}$
- 2nd row is $L_{2,1}$
- 3rd row is $L_{2,2}$

```
>>L(1,:) = poly([2 2.5])/((1-2)*(1-2.5))
L=      0.6667      -3.0000     3.3333

>>L(2,:) = poly([1 2.5])/((2-1)*(2-2.5));
L=      0.6667      -3.0000     3.3333
        -2.0000     7.0000      -5.0000

>>L(3,:) = poly([1 2])/((2.5-1)*(2.5-2));
L=      0.6667      -3.0000     3.3333
        -2.0000     7.0000      -5.0000
        1.3333      -4.0000     2.6667
```

Lagrange Coefficients Polynomial P computed as

```
>>P=3*L(1,:) + 3*L(2,:) + 3.3*L(3,:)
P=0.4000       -1.2000          3.8000
```

The **pretty** function is used to display the formation polynomial output

```
>>pretty(poly2sym(P)) ;
```

Evaluate the polynomial at 1.5

```
>>polyval(P,1.5)
ans=2.9000
```

Comparing the true value of f(x)=x+2/x at 1.5

```
>>1.5+2/1.5
ans=2.8333
```

Exercise 1

Verify above coefficient for the x=1 , x=1.2,   x=1.7

Plot both of these on a graph to compare

```
>>SP=poly2sym(P)
SP=2/5*x^2-6/5*x+19/5

>>ezplot('x+2/x',[0.5 3]
>>hold on
>>ezplot(SP,[0 3]
```

Exercise 2

Calculate the third degree Lagrange polynomial for f(x)=cos(x). X=[0.0 0.4 0.8 1.2] Verify output for two different value of x and also plot the graph for comparison.

Exercise 3

Calculate interpolation polynomial for given point X= [0:5] and function f(x) =1.5^x cos(2x), by using Lagrange polynomial. Make a figure for Lagrange polynomial. In this figure show the points, a plot of function, and curve that corresponds to this method.

Pseudo code for Lagrange Coefficient

procedure (C: lagan coefficients, L: lagran poly) lagran (X: given points, Y: function values)

```
begin
for each k:1 to n+1 do
    V=1
    for each j=1 to n+1 do
        if k not equal j then
        V=conv(V,poly(X(j)))/(X(k)-X(j))
        end
    end
    Store Lagrange coefficient in variable L
end
C=Y*L
end
```

Exercise 4

Write the Matlab function with name lagran to compute the generalized Lagrange coefficient and also display the graph of comparison.