# Department of Computing

# MATH 333: Numerical Analysis

# Class: BSCS-9ABC

# Lab 6: Polynomial, Interpolation, Curve fitting (Part 1)

# Date: March 18, 2022

# Time: 10:00 pm-1:00 pm & 2:00 pm – 5:00 pm

# Lab Engineer: Anum Asif

|  |
| --- |
| **Submitted By:**  **Fatima Seemab**  **291310**  **BSCS 9B** |

# Lab 6: Polynomial,Interpolation

**Introduction**

Matlab represents polynomials with a vector of coefficients. The length of the vector will always be one more than the order of the polynomial

y = x2 + 5x -3 is represented as [1 5 -3]

y = -x3/3 + x is represented as [-1/3 0 1 0]

**Objectives**

The purpose of this lab is to get familiar with Interpolation.

**Tools/Software Requirement**

Matlab R2016a

**Description**

**Interpolation**

Given a set of data points, you can use use **interp1.m** to interpolate between them at one or more x values. The method defaults to a linear interpolation.

x = [0 2 4];

y = [0 2 8];

plot(x,y,'sk-')

hold on

x2 = [1 3]

y2linear = interp1(x,y,x2)

plot(x2,y2linear,'go')

legend('Data','Linear')

**Curve Fitting with polynomials**

If you have a set of data points that you want to find a 'best fit line' for, use the **polyfit** command. This will perform a 'least squares' fit which minimizes the error (distances) between the data and the fitted curve. The experimental data will not necessarily lie on the fitted curve. The **polyval** command is used to evaluate a polynomial for a given set of x values. This is useful in interpolation and plotting.

**Find the equation of the line that best fits a set of data**

data=[0 .3

1 .9

2 2.2

3 3.1

4 4.0

5 5.4];

x=data(:,1);

y=data(:,2);

% get polynomial coefficients (pc) of bets fit line

pc = polyfit(x,y,1); % 1 means linear fit

plot(x,y,'ro');

hold on;

plot(x,polyval(pc,x),'b-');

txt = sprintf('Best fit line y=%.2fx + %.2f',pc(1),pc(2));

legend('Data points',txt);

**Lab Task**

Calculate and plot a best fit line for this data set.

t=[0:8]; y=[40.12 66.78 80.17 86.71 80.77 66.78 44.41 10.51 -32.60];

Find:

1. What is the slope and intercept?

**CODE:**

|  |
| --- |
| clear  clc  data=[40.12 66.78 80.17 86.71 80.77 66.78 44.41 10.51 -32.60];  x=(0:8);  % get polynomial coefficients (pc) of bets fit line  pc = polyfit(x,data,1); % 1 means linear fit  plot(x,data,'ro');  hold on;  plot(x,polyval(pc,x),'b-');  txt = sprintf('Best fit line y=%.2fx + %.2f',pc(1),pc(2));  legend('Data points',txt);  fprintf("The slope is:"+pc(1)+"\n")  fprintf("The y intercept is:"+pc(2)+"\n") |

**OUTPUT:**

|  |
| --- |
| Chart  Description automatically generated |

**Slope and y intercept:**

|  |
| --- |
|  |

1. What is the value of y when t=4.5 and t=8.5

**CODE:**

|  |
| --- |
| clear  clc  data=[40.12 66.78 80.17 86.71 80.77 66.78 44.41 10.51 -32.60];  x=(0:8);  pc = polyfit(x,data,1);  plot(x,data,'ro');  val1 = polyval((pc),4.5);  val2 = polyval((pc),8.5);  hold on;  plot(x,polyval(pc,x),'b-');  plot(val1,data,'bo');  txt = sprintf('Best fit line y=%.2fx + %.2f',pc(1),pc(2));  legend('Data points',txt);  fprintf("The value at 4.5 is:"+val1+"\n")  fprintf("The value at 8.5 is:"+val2+"\n") |

**OUTPUT:**

|  |
| --- |
|  |

1. Use interp1 for part 2 - why the different values

**CODE:**

|  |
| --- |
| clear  clc  data=[40.12 66.78 80.17 86.71 80.77 66.78 44.41 10.51 -32.60];  x=0:8;  pc = polyfit(x,data,1);  plot(x,data,'ro');  val1 = polyval((pc),4.5);  val2 = polyval((pc),8.5);  hold on;  plot(x,polyval(pc,x),'b-');  txt = sprintf('Best fit line y=%.2fx + %.2f',pc(1),pc(2));  legend('Data points',txt);  fprintf("The value at 4.5 is:"+val1+"\n")  fprintf("The value at 8.5 is:"+val2+"\n")  xi=[4.5,8.5];  y1=interp1(x,data,xi(1));  y2=interp1(x,data,xi(2));  fprintf("The value at 4.5 using interp1 is:"+y1+"\n")  fprintf("The value at 8.5 using interp1 is:%d \n",y2)  plot(xi(1),y1,"b\*")  plot(xi(1),y2,"g\*") |

**OUTPUT:**

|  |
| --- |
| **Text  Description automatically generated** |

|  |
| --- |
| Chart  Description automatically generated |

**- why the different values**

|  |
| --- |
| The polyval function uses the values of m and c returned from polyfit method and put 4.5 and 8.5 in that function. It just finds the function coefficients and then put the value in the function. The interp1 function uses interpolation and finds the value at given points using data. It returns values at specific query points using linear interpolation |

**Deliverables**

Submit single word file with matlab code and screen shot of Output.