###### University of Dhaka

#### Department of Electrical and Electronic Engineering

EEE-3102: Numerical Technique Laboratory

Exp03: Determination of model parameters using polynomial curve fitting technique.

|  |  |  |
| --- | --- | --- |
| Section | : | A |
| Roll No.: | : | JN-19  SH-11 |
| Name | : | Sriman Bidhan Baray  Sharar Muhtasim |
| Date | : | 05-02-2020 |

**Theory for curve fitting:**

Curve fitting is a technique of finding an algebraic relationship that “best” (in least squares sense) fits a given set of data. Unfortunately, there is no magical function (in MATLAB or otherwise) that can give the relationship if we simply supply the data. We have to have an idea of what kind of relationship might exist between the input data and the output data. However, if we do not have the firm idea but have data that we trust, MATLAB can help us in exploring the best possible fit.

**Polynomial curve fitting:**

Polynomial model has the mathematical form given below:



Where, the power of *x* determined the order of the polynomial and *a0,* *a1, a2, a3,* etc is constants also called the parameters of the model. For example, if the order of the following polynomial is 2, it has three model parameters namely *a0,* *a1,* and *a2* and the polynomial equation is as follows:



Suppose we have a ‘*y*’ data set for a specific ‘*x*’ data set. Now, in polynomial curve fitting, we need to find the values of the model parameters (i.e., *a0,* *a1, a2,* etc).

**Linear regression**

The simplest example of linear regression is fitting a straight line to a set of paired observations: (x0, y0), (x1, y1),…… (xn, yn). The mathematical expression for straight line is

y=a0+a1x

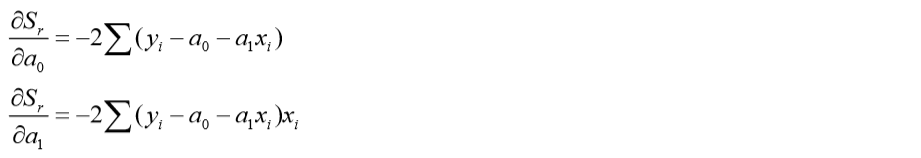
Where a0 and a1 are coefficients representing the intercept and slope (model parameters) and y is the model value. The order of the polynomial is n = 1. If yi,observed is the observed value and yi,model is the modeled value then error (residual) ei for the ith value is given by



For a total of ‘n’ observation, there will be ‘n’ error terms (residuals). Now we need some criteria such that the error is minimum. One such strategy is to minimize the sum of the square errors which is given by



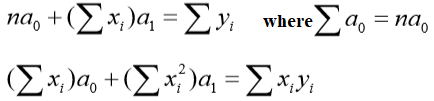
To determine the values of a0 and a1, the above equation is differentiated with respect to each coefficient to obtain



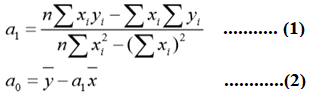
Setting these derivatives equal to zero will result in a minimum Sr. If this is done, the equations can be expressed as



We can express the above equations as a set of two simultaneous linear equations with two unknowns’ a0 and a1 as follows

****

Solving above two equations, we have



Similar expressions can be obtained for polynomial equation with higher order also.

**Q.1:** suppose x values are within the range of  and y = 2+3x. Determine the model parameters using the above Eq. (1) and Eq. (2). Note that, for example, ∑x is done by ‘sum(x)’ and  is done by ‘mean(x)’ in Matlab.

1. **Paste your code below**

**Code:**

clc; close all; clear all;

x = 0:0.1:10;

y=2+3\*x;

n = length(x);

term\_1 = n\*(sum(x.\*y));

term\_2 = sum(x)\*sum(y);

x\_sq = x.^2;

term\_3 = n\*sum(x\_sq);

term\_4 = (sum(x)).^2;

y\_bar = mean(y);

x\_bar = mean(x);

a1 = (term\_1 - term\_2)/(term\_3 - term\_4)

a0 = y\_bar - (a1\*x\_bar)

1. **What are the model parameters obtained by running your code?**

**Answer:**

a1 = 3.0000

a0 = 2.0000

**Example-1:** Observed that y0 = 2+3x for  is a polynomial equation and the order of the equation is n =1. Now you can use matlab built-in function ‘polyfit’ to perform the same job. To use this function, try the following code

clc;close all;

x=0:0.5:10;

y0=2\*x+3; %% let it gives observed values

n=1; % order used

a=polyfit(x,y0,n) %% gives model parameters

ym=polyval(a,x); % gives the fittd values

MSE=mean((y0-ym).^2); %gives mean square error

figure;

plot(x,y0,'-bs')

hold on

plot(x,ym,'-ro')

grid on

legend('observed','modelled')

**Q.2: Comment on the curve you obtained after running the code in example-1. What is the value of mean square error (MSE)? Why MSE value is so small?**

**Answer:**

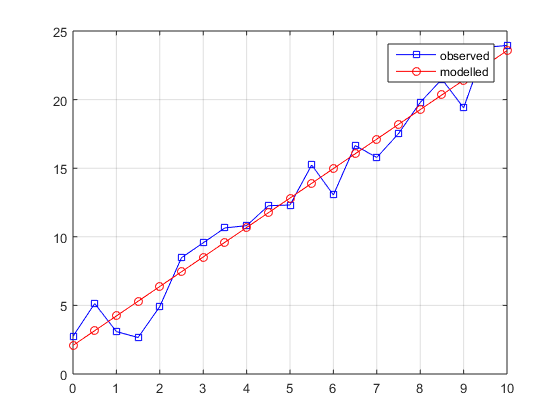
The modeled curve and the observed curve were exactly same as we got the data from the same equation we got after the modelling.

MSE = 6.3203e-30

The value of MSE is very small because there was no error. In fact the modeled and observed curve were exactly same.

**Q.3:** Given that that y = 2+3x for  is a polynomial equation (linear in this case). Produce yo by adding noise to y such that the SNR = 20dB. Now use the same procedures of **Example-1** to fit yo.

1. **Plot the observed data yo and modeled data ym in the same plot and paste the plot below.**

****

**(ii) What are the model parameters now?**

**Ans:**

a =

2.1483 2.0824

1. **What is the value of MSE now?**

MSE =

1.6284

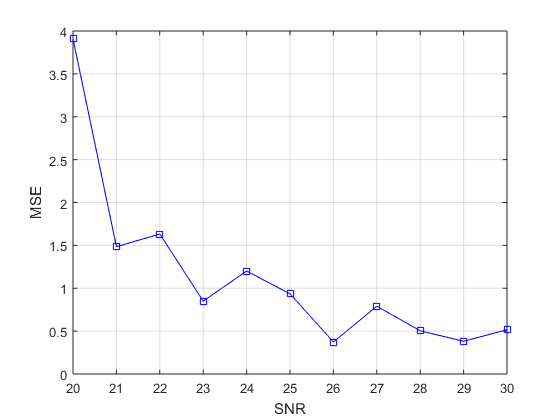
1. **List the model parameters for SNR of 25, 30 and 35 dB and comment on the values of the parameters.**

Ans: snr = 25, a =1.9888 3.1559

Snr=30, a =2.0135 2.9111

Snr = 35, a = 2.0280 2.7833

As the snr increases, value of parameters get closer to actual values of the parameters of y.

**(iv) Run the code for SNR from 20 to 30 dB with 1 dB step each time and store the MSE in an excel file. Draw a curve to show the plot of SNR (dB) versus MSE. **

**Q.4:** For  with an increment of 0.02, given that



1. **fit the data y with 10th order ( n= 10) polynomial and find the model parameters and MSE. Comment on your results.**

Answers:

a =

Columns 1 through 6

10.0000 9.0000 8.0000 7.0000 6.0000 5.0000

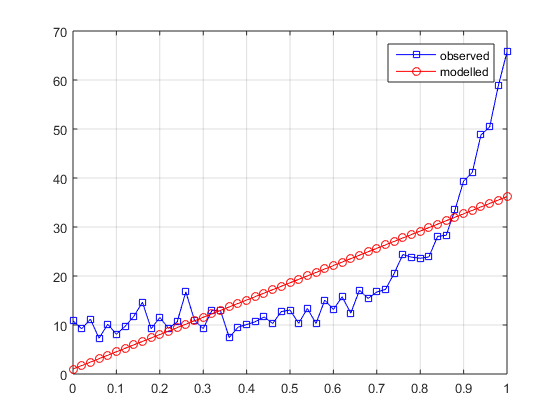
Columns 7 through 11

4.0000 3.0000 2.0000 1.0000 10.0000

MSE =

2.6790e-29

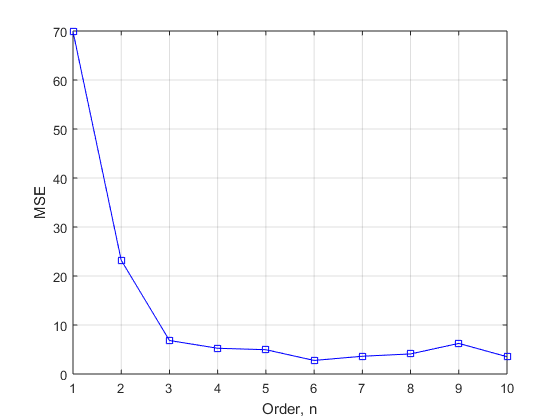
1. **Now, Produce yo by adding noise to y such that the SNR = 20dB. Now use the same procedures of Example-1 to fit yo. What is the value of MSE for n =1 (i.e., first order polynomial)?**

****

MSE =

69.2814

1. **Vary the order n of the polynomial from 1 to 10 and record your result in an excel file. Draw a plot to show a graph of order (n) versus MSE.**

****

**The End**

**References**

[1] S.K. Mitra, Digital Signal Processing, 3rd Edition, McGraw-Hill Education (Asia), 2009.

[2] J.G. Proakis and D.G. Manolakis, Digital Signal Processing: Principles, Algorithms and Applications, 4th Edition, Pearson International Edition, 2007.

[3] McClellan, Schafer and Yoder, *DSP FIRST: A Multimedia Approach*. Prentice Hall, Upper Saddle River, New Jersey, 1998 Prentice Hall.

[4] *Using Matlab*, The Math Works Inc.