ECE 8540 Analysis of tracking systems

Lab 2 - Nonlinear Regression and Model Fitting

Rahil Modi
C14109603

September 8, 2020

1 Introduction

In this Lab we are asked to develop a code to calculate the non linear regression fit for a function of the form $y = ln(ax)$.

2 Methods

In this section I will be doing the mathematical derivation for fitting a model of the form $y = ln(ax)$ to the three data files provided where a is the unknown. The error function is:

$$E = \sum(Data - Model) \tag{1}$$

The error function as per our function of form:

$$E = \sum_{n=1}^{N}(y_i - ln(a_i x))^2 \tag{2}$$

After taking the partial derivative wrt to the unknown a:

$$\frac{\partial E}{\partial a} = \sum_{n=1}^{N}(y_i - ln(a_i x)) * (-\frac{2}{a}) \tag{3}$$

Equating it to zero:

$$\frac{\partial E}{\partial a} = \sum_{n=1}^{N}(y_i - ln(a_i x)) * (-\frac{2}{a}) = 0 \tag{4}$$

$$= \sum_{n=1}^{N}(\frac{2}{a})(y_i - ln(a_i x)) = 0 \tag{5}$$

$$= \sum_{n=1}^{N}(\frac{1}{a})(y_i - ln(a_i x)) = 0 \tag{6}$$

To solve this problem iteratively using the root finding approach, let:

$$f(a) = \sum_{n=1}^{N} (\frac{1}{a})(y_i - ln(a_i x)) \tag{7}$$

$$= \sum_{n=1}^{N} (\frac{y_i}{a}) - (\frac{ln(a_i x)}{a}) \tag{8}$$

The derivative of this function f'(a) is :

$$f'(a) = \frac{\partial f(a)}{\partial a} = \sum_{n=1}^{N} (-\frac{y_i}{a^2}) - (\frac{-ln(a_i x)}{a^2} + (\frac{1}{a}) * (\frac{x_i}{a x_i})) \tag{9}$$

$$f'(a) = \sum_{n=1}^{N} (\frac{-y_i - 1 - ln(a_i x)}{a^2}) \tag{10}$$

After taking an initial guess, successive iterations of the following converge to a solution:

$$a_n + 1 = a_n - (\frac{f(a_n)}{f'(a_n)}) \tag{11}$$

2.1 Data Set 1

We were given a data set named log-data-A.txt using which we have to fit a model through these data points.
The MATLAB code for the same is given below:

```
%% Log Data A
A = importdata('log-data-A.txt') ;
figure;
scatter (A(:, 1), A(:, 2), 40, 'b.');
title ('Scatter Plot of Log Data A');
xlabel('X datapoints');
ylabel('Y datapoints');
```

```
fa = 0;
fan = 0;
an = 7 ;                                                    %initial guess
X = ['The estimated initial guess is ', num2str(an),'.'];
disp (X);
iter = 0;
for i =1:100
fa(i) = sum((A(:, 2) - log(an.*A(:, 1)))./an);
fan(i) = sum((log(an.*A(:, 1)) - 1 - A(:, 2))./an ^2);
an1 = an - (fa(i)./fan(i));
if(abs(an1 - an)<0.000001)
iter = i ;
break ;
end
an = an1 ;
end
Y = ['The final value of a is ', num2str(an1), '.'];
disp (Y)
Z = ['The number of iterations it took to reach is ', num2str(iter), '.'];
disp (Z)
x = 0 : 50;
y = log(an.*x);
figure
hold on
g = plot(x, y, 'r');
h = plot(A(:, 1), A(:,2), 'b.');
uistack(g, 'top');
title ('Curve Fitting of Log Data A');
xlabel('X datapoints');
ylabel('Y datapoints');
legend ('Data Points', 'Fitted Curve');
hold off
```

2.2 Data set 2

We were given a data set named log-data-B.txt using which we have to fit a model through these data points.
The MATLAB code for the same is given below:

```
%% Log Data B
B = importdata('log-data-B.txt') ;
figure;
scatter (B(:, 1), B(:, 2), 40, 'k.');
title ('Scatter Plot of Log Data B');
xlabel('X datapoints');
ylabel('Y datapoints');
fa_2 = 0;
fan_2 = 0;
an_2 = 19 ;                                                    %initial guess
D = ['The estimated initial guess is ', num2str(an_2),'.'];
disp (D);
iter = 0;
for i =1:100
fa_2(i) = sum((B(:, 2) - log(an_2.*B(:, 1)))./an_2);
fan_2(i) = sum((log(an_2.*B(:, 1)) - 1 - B(:, 2))./an_2 ^2);
an1_2 = an_2 - (fa_2(i)./fan_2(i));
if(abs(an1_2 - an_2)<0.000001)
iter = i ;
break ;
end
an_2 = an1_2 ;
end
E = ['The final value of a is ', num2str(an1_2), '.'];
disp (E)
F = ['The number of iterations it took to reach is ', num2str(iter), '.'];
disp (F)
x = 0 : 50;
y = log(an_2.*x);
figure
hold on
g = plot(x, y, 'g');
h = plot(B(:, 1), B(:,2), 'k.');
uistack(g, 'top');
title ('Curve Fitting of Log Data B');
xlabel('X datapoints');
ylabel('Y datapoints');
legend ('Data Points', 'Fitted Curve');
```

```
hold off
```

2.3 Data set 3

We were given a data set named log-data-C.txt using which we have to fit a model through these data points.
The MATLAB code for the same is given below:

```
%% Log Data C
C = importdata('log-data-C.txt') ;
figure;
scatter (C(:, 1), C(:, 2), 40, 'm.');
title ('Scatter Plot of Log Data C');
xlabel('X datapoints');
ylabel('Y datapoints');
fa_3 = 0;
fan_3 = 0;
an_3 = 0.30;                                                    %initial guess
G = ['The estimated initial guess is ', num2str(an_3),'.'];
disp (G);
iter = 0;
for i =1:100
fa_3(i) = sum((C(:, 2) - log(an_3.*C(:, 1)))./an_3);
fan_3(i) = sum((log(an_3.*C(:, 1)) - 1 - C(:, 2))./an_3^2);
an1_3 = an_3 - (fa_3(i)./fan_3(i));
if(abs(an1_3 - an_3)<0.000001)
iter = i ;
break ;
end
an_3 = an1_3 ;
end
H = ['The final value of a is ', num2str(an1_3), '.'];
disp (H)
I = ['The number of iterations it took to reach is ', num2str(iter), '.'];
disp (I)
x = 0 : 50;
y = log(an_3.*x);
figure
```

```
hold on
g = plot(x, y, 'c');
h = plot(C(:, 1), C(:,2), 'm.');
uistack(g, 'top');
title ('Curve Fitting of Log Data C');
xlabel('X datapoints');
ylabel('Y datapoints');
legend ('Data Points', 'Fitted Curve');
hold off
```

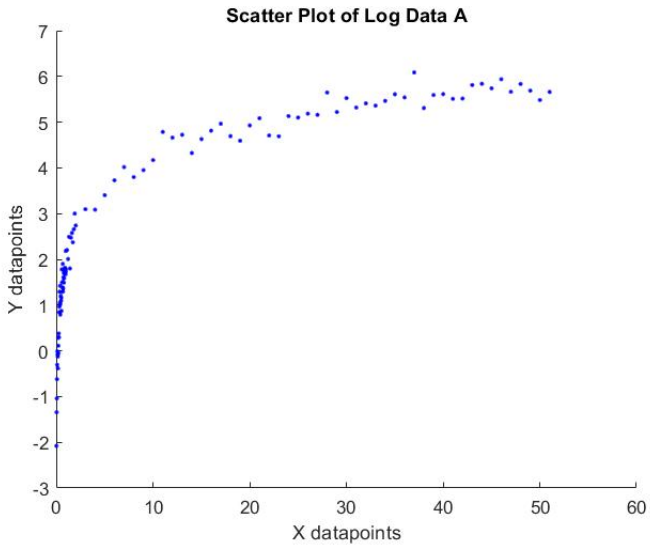
3 Results

3.1 Data set 1

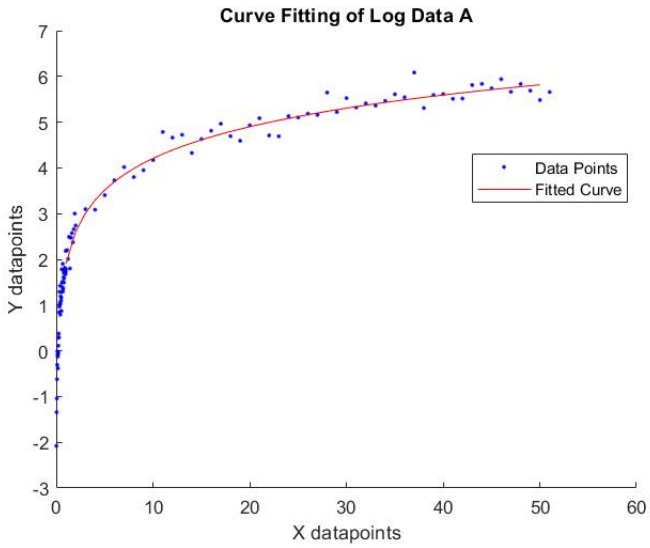
3.1.1 Output in command window

```
The estimated initial guess is 7.
The final value of a is 6.7114.
The number of iterations it took to reach is 4.
```

3.1.2 Plot of data points and curve fitting



Scatter plot of the data set 1

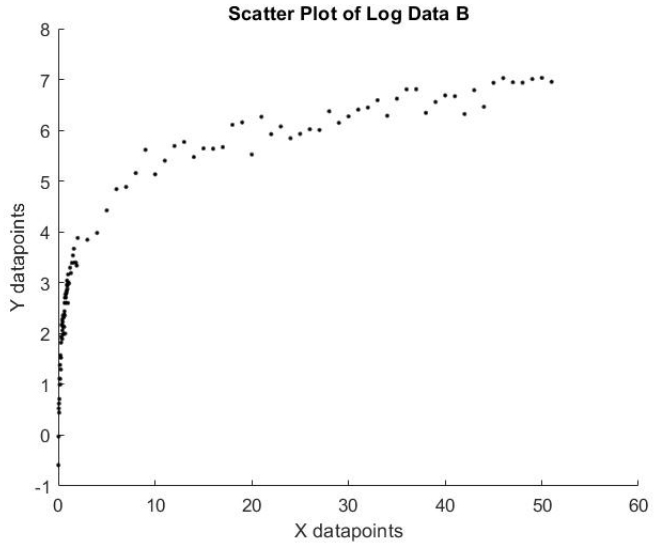


Fitted curve of the data set 1

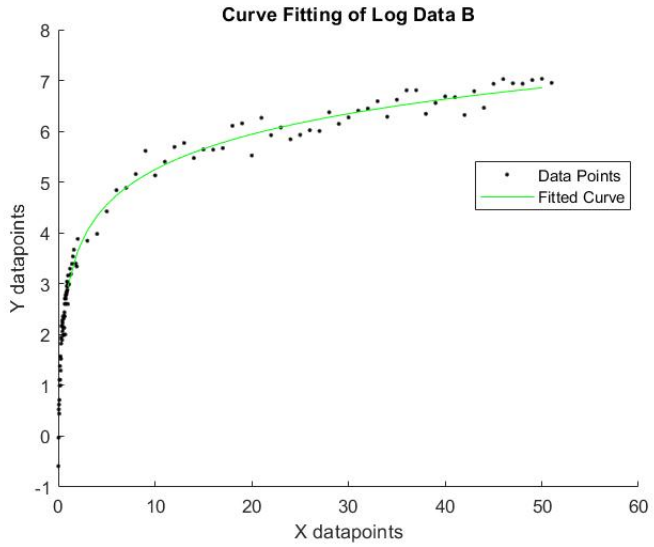
3.2 Data set 2

```
The estimated initial guess is 19.
The final value of a is 18.9961.
The number of iterations it took to reach is 3.
```

3.2.1 Plot of data points and curve fitting



Scatter plot of the data set 2



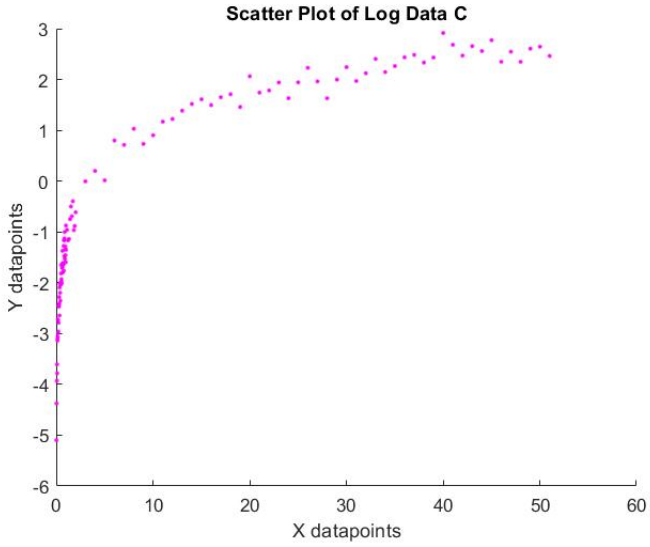
Fitted curve of the data set 2

3.3 Data set 3

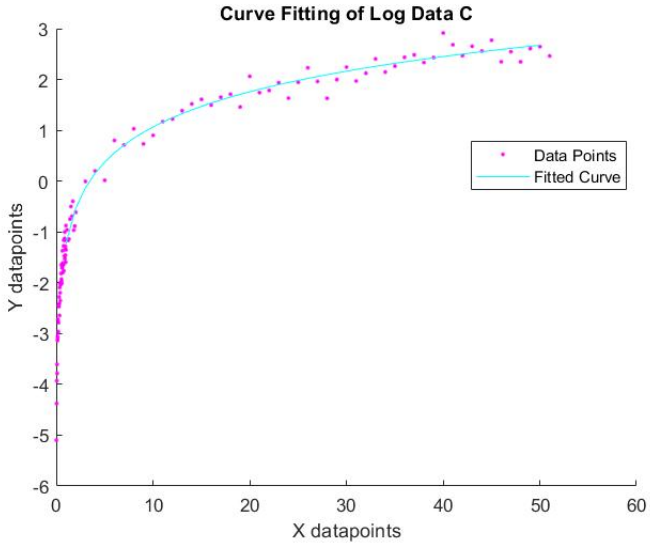
```
The estimated initial guess is 0.3.
The final value of a is 0.29.
```

```
The number of iterations it took to reach is 4.
```

3.3.1 Plot of data points and curve fitting



Scatter plot of the data set 3



Fitted curve of the data set 3

4 Conclusion

We have learned in this Lab how to fit a non linear regression curve through a form of function. We also learned the root finding method for the model.

5 Appendix

The entire MATLAB code is given below:

```
clc
clear all
close all
%% Log Data A
A = importdata('log-data-A.txt') ;
figure;
scatter (A(:, 1), A(:, 2), 40, 'b.');
title ('Scatter Plot of Log Data A');
xlabel('X datapoints');
ylabel('Y datapoints');
fa = 0;
fan = 0;
an = 7 ;                                                    %initial guess
X = ['The estimated initial guess is ', num2str(an),'.'];
disp (X);
iter = 0;
for i =1:100
fa(i) = sum((A(:, 2) - log(an.*A(:, 1)))./an);
fan(i) = sum((log(an.*A(:, 1)) - 1 - A(:, 2))./an ^2);
an1 = an - (fa(i)./fan(i));
if(abs(an1 - an)<0.000001)
iter = i ;
break ;
end
an = an1 ;
end
Y = ['The final value of a is ', num2str(an1), '.'];
```

```
disp (Y)
Z = ['The number of iterations it took to reach is ', num2str(iter), '.'];
disp (Z)
x = 0 : 50;
y = log(an.*x);
figure
hold on
g = plot(x, y, 'r');
h = plot(A(:, 1), A(:,2), 'b.');
uistack(g, 'top');
title ('Curve Fitting of Log Data A');
xlabel('X datapoints');
ylabel('Y datapoints');
legend ('Data Points', 'Fitted Curve');
hold off
%% Log Data B
B = importdata('log-data-B.txt') ;
figure;
scatter (B(:, 1), B(:, 2), 40, 'k.');
title ('Scatter Plot of Log Data B');
xlabel('X datapoints');
ylabel('Y datapoints');
fa_2 = 0;
fan_2 = 0;
an_2 = 19 ;                                                    %initial guess
D = ['The estimated initial guess is ', num2str(an_2),'.'];
disp (D);
iter = 0;
for i =1:100
fa_2(i) = sum((B(:, 2) - log(an_2.*B(:, 1)))./an_2);
fan_2(i) = sum((log(an_2.*B(:, 1)) - 1 - B(:, 2))./an_2 ^2);
an1_2 = an_2 - (fa_2(i)./fan_2(i));
if(abs(an1_2 - an_2)<0.000001)
iter = i ;
break ;
end
an_2 = an1_2 ;
end
```

```
E = ['The final value of a is ', num2str(an1_2), '.'];
disp (E)
F = ['The number of iterations it took to reach is ', num2str(iter), '.'];
disp (F)
x = 0 : 50;
y = log(an_2.*x);
figure
hold on
g = plot(x, y, 'g');
h = plot(B(:, 1), B(:,2), 'k.');
uistack(g, 'top');
title ('Curve Fitting of Log Data B');
xlabel('X datapoints');
ylabel('Y datapoints');
legend ('Data Points', 'Fitted Curve');
hold off
%% Log Data C
C = importdata('log-data-C.txt') ;
figure;
scatter (C(:, 1), C(:, 2), 40, 'm.');
title ('Scatter Plot of Log Data C');
xlabel('X datapoints');
ylabel('Y datapoints');
fa_3 = 0;
fan_3 = 0;
an_3 = 0.30;                                                    %initial guess
G = ['The estimated initial guess is ', num2str(an_3),'.'];
disp (G);
iter = 0;
for i =1:100
fa_3(i) = sum((C(:, 2) - log(an_3.*C(:, 1)))./an_3);
fan_3(i) = sum((log(an_3.*C(:, 1)) - 1 - C(:, 2))./an_3^2);
an1_3 = an_3 - (fa_3(i)./fan_3(i));
if(abs(an1_3 - an_3)<0.000001)
iter = i ;
break ;
end
an_3 = an1_3 ;
```

```
end
H = ['The final value of a is ', num2str(an1_3), '.'];
disp (H)
I = ['The number of iterations it took to reach is ', num2str(iter), '.'];
disp (I)
x = 0 : 50;
y = log(an_3.*x);
figure
hold on
g = plot(x, y, 'c');
h = plot(C(:, 1), C(:,2), 'm.');
uistack(g, 'top');
title ('Curve Fitting of Log Data C');
xlabel('X datapoints');
ylabel('Y datapoints');
legend ('Data Points', 'Fitted Curve');
hold off
```