ECE 8540 Analysis of Tracking Systems
Assignment 1

Vivek Koodli Udupa
C12768888

09/04/2018

1 Aim

Part 1:

To fit a 2D line to the given five (x , y) data points. Construct the appropriate matrices and solve for the line parameters using the Normal Equations.

Part 2:

Follow the same steps as Part 1 but include an extra point (8 , 14). Find out what happens.

Part 3:

Fit a model for the data set that contains the data of 3398 meals eaten by 83 different people. Using the normal equations, formulate the matrices and fit that model to the data.

2 Execution

Part 1 and Part 2

The Dataset for Part 1: (5 , 1); (6 , 1); (7 , 2); (8 , 3); (9 , 5).
The Dataset for Part 2: (5 , 1); (6 , 1); (7 , 2); (8 , 3); (9 , 5); (8 , 14).

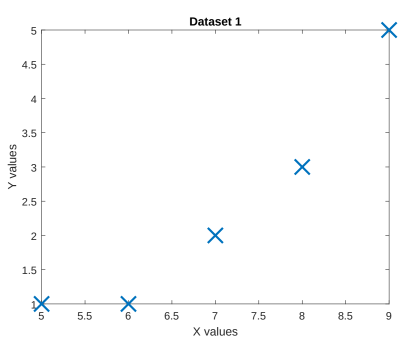

Figure 1: Plot of Dataset 1

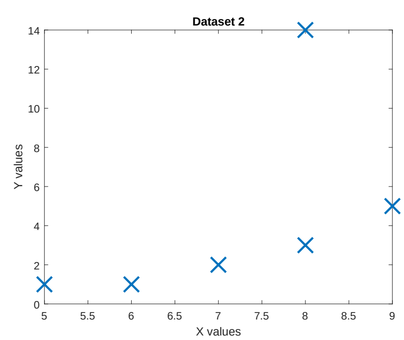

Figure 2: Plot of Dataset 2

Looking at the plot in Figure 1 and Figure 2, a straight line of the form y = ax + b seems to be a good fit.

Formulating the matrices:

$$A = \begin{bmatrix} x_1 & 1 \\ .. & .. \\ .. & .. \\ x_N & 1 \end{bmatrix} \tag{1}$$

$$x = \begin{bmatrix} a \\ b \end{bmatrix} \tag{2}$$

$$b = \begin{bmatrix} y1 \\ .. \\ .. \\ y_N \end{bmatrix} \tag{3}$$

A is a N x M Matrix,
where N = 5 (Number of data points) and M = 2 (Number of unknowns)

Filling in the matrices with appropriate values for Data set 1, we get:

$$A = \begin{bmatrix} 5 & 1 \\ 6 & 1 \\ 7 & 1 \\ 8 & 1 \\ 9 & 1 \end{bmatrix}$$

$$b = \begin{bmatrix} 1 \\ 1 \\ 2 \\ 3 \\ 5 \end{bmatrix}$$

$$x\_u = \begin{bmatrix} a \\ b \end{bmatrix}$$

With the above matrices, we can find x_u as :

$$x\_u = (A^T A)^{-1} A^T b \tag{4}$$

Using MATLAB for calculations, we get

$$x\_u = \begin{bmatrix} -1.00 \\ -4.60 \end{bmatrix}$$

Similarly formulating the matrices for data-set 2, we get:

$$A2 = \begin{bmatrix} 5 & 1 \\ 6 & 1 \\ 7 & 1 \\ 8 & 1 \\ 9 & 1 \\ 8 & 14 \end{bmatrix}$$

$$b2 = \begin{bmatrix} 1 \\ 1 \\ 2 \\ 3 \\ 5 \\ 14 \end{bmatrix}$$

$$x\_u2 = \begin{bmatrix} a \\ b \end{bmatrix}$$

Using MATLAB for calculations, we get

$$x\_u2 = \begin{bmatrix} 1.8154 \\ -8.6769 \end{bmatrix}$$

Now, we can fit a line of the form

$$y = a * x + b; \quad \text{where } a = x\_u(1,1) \text{ and } b = x\_u(2,1)$$

(Note: Similar line fit is calculated for data set 2 with its respective values)

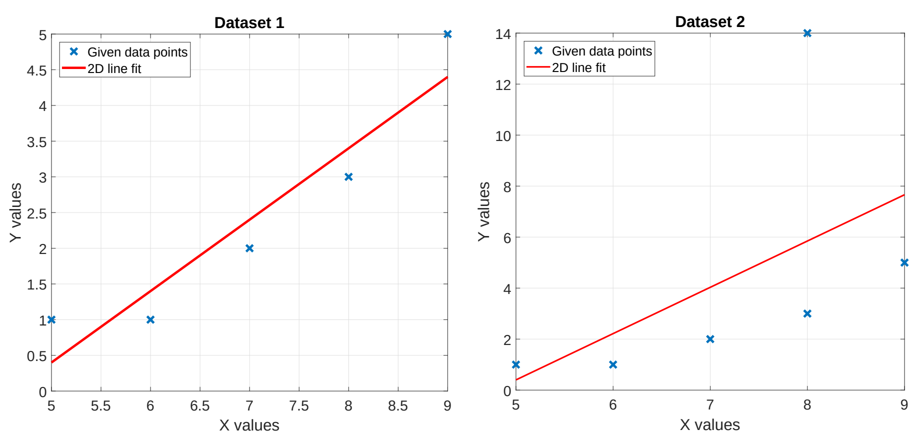

Figure 3: Dataset 1 and Dataset 2 Line Fit

Figure 3 shows the dataset plot with fitted line.

Part 3

The data for part 3 of the lab contains 3,398 meals eaten by 83 different people. It has 4 columns and they represent the following:

1. Participant ID
2. Meal ID
3. Number of Bites taken in the meal
4. Number of Kilo calories consumed

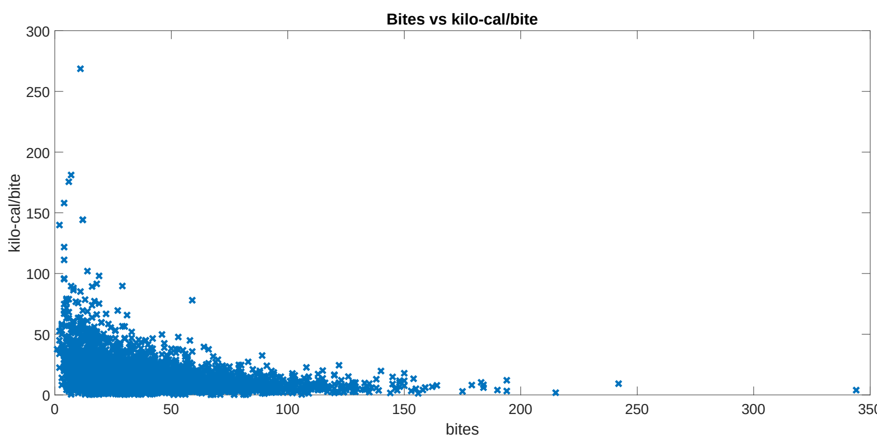

Figure 4: Plot of bites VS Kilo-calories per bite

Figure 4 shows the plot of kilo-calories consumed per bite against the total number of bites taken.
The values of the Y-axis is obtained by dividing Column 4 / Column 3 from the given data-set. The values of X-axis are from Column 3 of the Data-set.

The distribution of the given data as shown in Figure 4 is concentrated near the origin and scarce at the farther end of X and Y axes.

We will try to fit a line using the same methods as we used for Part 1 and Part 2, we get:

$$x\_u3 = \begin{bmatrix} -0.1771 \\ 23.4417 \end{bmatrix}$$

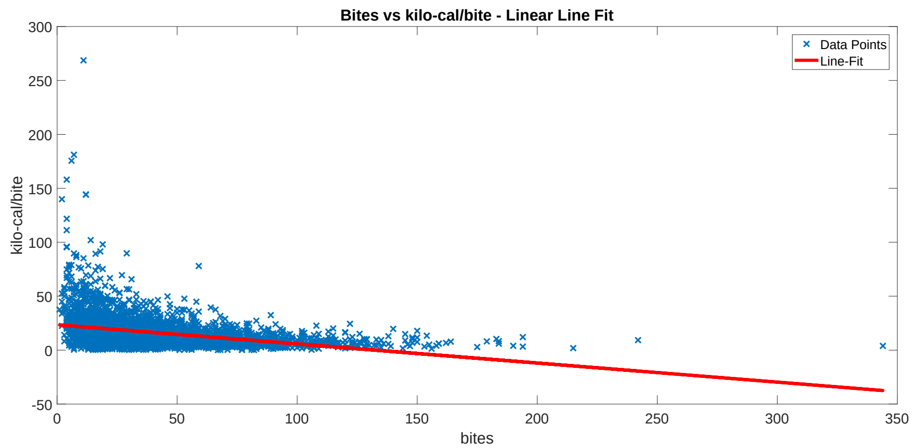

Figure 5: Line fit for Part 3 Data-set

Fitting a line using the values obtained gives us the result shown in Figure 5. The line extends way ahead along the X-axis and has high error values for points in the Y-axis. The linear line is not a great fit for this data set.

Power fit with the equation of the form $y = ax^b$ seems to be a better option. Here we have two unknowns (a and b). We have to model the above equation.

$$y = ax^b \tag{5}$$

$$log(y) = log(ax^b) \tag{6}$$

$$log(y) = log(a) + log(x^b) \qquad \because [log(mn) = log(m) + log(n)] \tag{7}$$

$$log(y) = log(a) + blog(x) \qquad \because [log(m^n) = nlog(m)] \tag{8}$$

Above equation is of the form

$$v = k + bu \qquad \text{where} \tag{9}$$

$$v = log(y) \tag{10}$$

$$k = log(a) \tag{11}$$

$$u = log(x) \tag{12}$$

Formulating the Matrices:

$$A = \begin{bmatrix} log(x_1) & 1 \\ .. & .. \\ .. & .. \\ log(x_N) & 1 \end{bmatrix} \tag{13}$$

$$x = \begin{bmatrix} b \\ a \end{bmatrix} \tag{14}$$

$$b = \begin{bmatrix} log(y1) \\ .. \\ .. \\ log(y_N) \end{bmatrix} \tag{15}$$

Using the solution to the normal equations i.e.

$$x = (A^T A)^{-1} A^T b \tag{16}$$

We get the values of the unknowns as a = 57.7128 and b = -0.4601. Now we can plot the power fit.

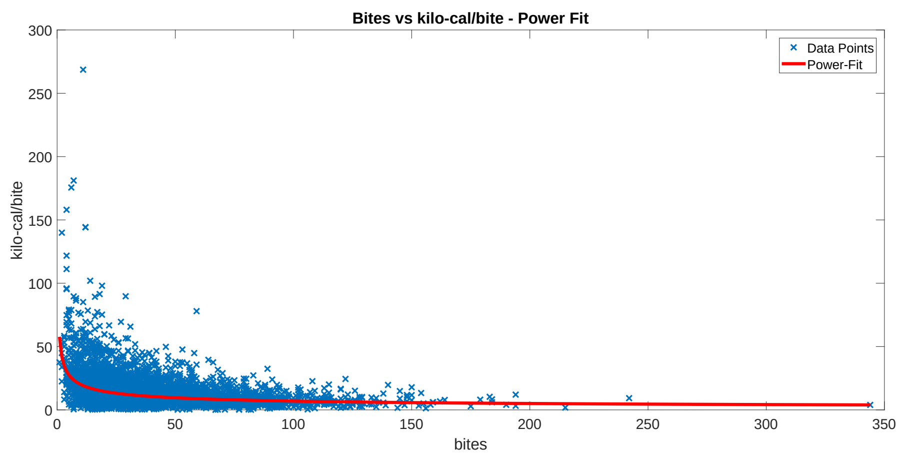

Figure 6: Power Fit

3 Conclusion

Part 1 and Part 2

The raw data-points given for Part 1 and Part 2 of the lab were plotted using MATLAB. Appropriate matrices for solving the line parameters using Normal Equations were constructed an the line fit was plotted. Figure 1 and Figure 2 represents the raw data plot and Figure 3 shows the line fit.
Due to the addition of an extra point (8 , 14), which has a high y-axis value, the y-intercept of the line fit approximately doubled. The unknowns for Part 1 are **a = -1.00 and b = -4.60** and the unknowns for Part 2 are **a = 1.8154 and b = -8.6769**.

Part 3

For Part 3 of the lab, the data-set named "83people-all-meals.txt" which was given in the website was imported into MATLAB as a table and then converted into array using the inbuilt function "table2array". Having the data as an array eases the calculation process. Kilo-calories per bite VS bites taken were plotted. Line model was used initially to fit the data. It did not fit well. Later on Power model of the form $y = ax^b$ was used, which seems to be a better fit than the Line model. Figure 5 and Figure 6 represent the Line model and power model fit respectively.

All calculations were performed on MATLAB and the code has been attached in the appendix.

4 Appendix

```
close all
clear

%Dataset one
px1 = [5, 6, 7, 8, 9];
py1 = [1, 1, 2, 3, 5];

%M = 2, N = length(x)

A1 = [px1' , ones(length(px1),1)]; %A Matrix
b1 = py1'; %Leftovers

x1 = inv(A1' * A1) * A1' * b1 %Calculating the unknowns

Y1 = x1(1) * px1 + x1(2); %Final Y1


%Plotting the graph
figure(1)
subplot(1,2,1)
plot ( px1, py1, 'x', 'markers',12,'LineWidth',3 ); % Given dataset plot
hold on
plot ( px1 , Y1 , 'r-','LineWidth',3);%2d Line fit plot
hold off;
title('Dataset 1');
ylabel('Y values');
xlabel('X values');
```

```
legend('Given data points', '2D line fit','Location','northwest');
set(gca,'FontSize',18);

%Dataset 2
px2 = [5, 6, 7, 8, 8, 9];
py2 = [1, 1, 2, 3, 14, 5];

A2 = [px2' , ones(length(px2),1)];
b2 = py2';

x2 = inv(A2' * A2) * A2' * b2

Y2 = x2(1) * px2 + x2(2);

%Plotting the Graph
subplot(1,2,2)
plot ( px2, py2, 'x','markers',12,'LineWidth',3 ); %Given dataset plot
hold on
plot ( px2 , Y2 , 'r-','LineWidth',2); %2d Line fit plot
hold off;
title('Dataset 2');
ylabel('Y values');
xlabel('X values');
legend('Given data points', '2D line fit','Location','northwest');
set(gca,'FontSize',18);

%Part Three ( Power Function)

%Read the dataset from file into a table
Tble = readtable('83people-all-meals.txt');

%Convert the table into an array
A1 = table2array(Tble);

%Extract the 3rd column(i.e Bites taken)
Bites = A1(:,3);

%Calculate Kilocalories per bite
CalBite = A1(:,4) ./ A1(:,3) ;

%setting up the Matrices

% y3 = ax^b;
```

```
% V = K + bu3
% V = log(y3)
% K = log(a)
% u3 = log(x)

u3 = log(Bites);
v = log(CalBite);

A3 = [u3 ones(length(Bites),1)];
b3 = [v];

x3 = inv(A3' * A3) * A3' * b3;

a3 = exp(x3(2,1));
b3 = x3(1,1);

X3 = 0:max(Bites);
Y3 = a3 * X3.^b3;

% Plotting the graph
figure(3)
plot(Bites,CalBite, 'x','markersize',10,'linewidth',2);
hold on;
plot(X3,Y3,'r-','LineWidth',4);
hold off;
xlabel("bites");
ylabel("kilo-cal/bite");
title("Bites vs kilo-cal/bite - Power Fit");
set(gca,'FontSize',18);

%Part 3 - Polynomial

A4 = [Bites ones(length(Bites),1)];
b4 = [CalBite];
x4 = inv(A4' * A4) * A4' * b4;

X4 = 0:max(Bites);
Y4 = x4(1,1) .* Bites + x4(2,1);

% Plotting the graph
figure(4)
plot(Bites,CalBite, 'x','markersize',10,'linewidth',2);
hold on;
plot(Bites,Y4,'r-','LineWidth',4);
hold off;
xlabel("bites");
```

```
ylabel("kilo-cal/bite");
title("Bites vs kilo-cal/bite - Linear Line Fit");
set(gca,'FontSize',18);
```

-END-