Department of Chemical Engineering

ChE-101: Approaches to Chemical Engineering Problem Solving

MATLAB Tutorial VII

Linear Regression Using Least Square Method (last updated 5/18/06 by GGB)

Objectives:

These tutorials are designed to show the introductory elements for any of the topics discussed. In almost all cases there are other ways to accomplish the same objective, or higher level features that can be added to the commands below.

Any text below appearing after the double prompt (>>) can be entered in the Command Window directly or in an m-file.

The following topics are covered in this tutorial;

Introduction

Procedure to perform linear regression in Matlab

Solved Problem using Matlab (guided tour)

Solved Problem using Excel (guided tour)

Introduction:

Regression of data consists of getting a mathematical expression that best fits all the data. That is given a set of experimental data in which the dependent variable "y" is a function of "x", the intention of regression is to determine and expression for:

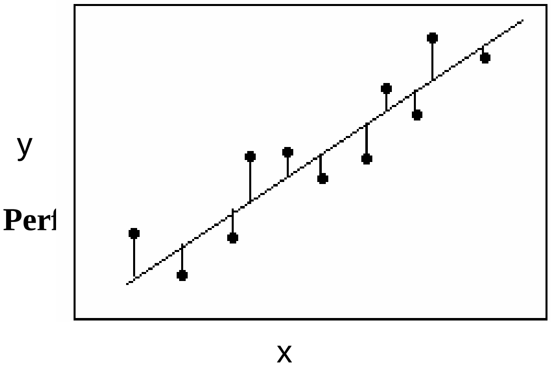
$$y = f(x) \tag{1}$$

For example, a set of experimental data could be predicted by using the following expression:

$$y = ax + b \tag{2}$$

The objective of regression is to determine the values for the parameters "a" and "b". Notice that in this case, the unknowns-the variables to calculate- are "a" and "b". Because the unknown variables (coefficients) are linear, the determination of the coefficients is known as "Linear Regression."

There are different methods to perform linear regression, the most common one is known as Least Square Method. As shown on the diagram below, the least squares method minimizes the sum of the squared distances between the points and the fitted line.



The sum of the squared distances from each point to the line are as small as possible.

Procedure to Perf

The objective is to determine the ‘m’ parameters 'a_1', 'a_2' and 'a_3' etc. in the equation,

$$y = a_1 x_1 + a_2 x_2 + \ldots . \; a_m$$

given a set of ‘n’ data points $(x_1, x_2, \ldots, x_{m-1}, y)$.

This is done by writing out the equation for each data point. This results in a set of ‘n’ equations in ‘m’ unknowns, a_1, a_2, a_3, ... , a_m

$$\begin{array}{lllllllll}
a_1 x_{1,1} & + & a_2 x_{2,1} & + & a_3 x_{3,1} & +\ldots. & a_m & = & y_1 \\
a_1 x_{1,2} & + & a_2 x_{2,2} & + & a_3 x_{3,2} & +\ldots. & a_m & = & y_2 \\
a_1 x_{1,3} & + & a_2 x_{2,3} & + & a_3 x_{3,3} & +\ldots. & a_m & = & y_3 \\
 & & & & : & & & & \\
a_1 x_{1,n} & + & a_2 x_{2,n} & + & a_3 x_{3,n} & +\ldots. & a_m & = & y_n
\end{array}$$

UNKNOWNS

where the first subscript on x identifies the independent variable and the second subscript signifies the data point

In matrix notation this is expressed as;

$$\begin{bmatrix} x_{1,1} & x_{2,1} & x_{3,1} & \ldots & 1 \\ x_{1,2} & x_{2,2} & x_{3,2} & \ldots & 1 \\ x_{1,3} & x_{2,3} & x_{3,3} & \ldots & 1 \\ & : & & \ldots & 1 \\ x_{1,n} & x_{2,n} & x_{3,n} & \ldots & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ : \\ a_m \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ : \\ y_n \end{bmatrix}$$

That is,

$$[\mathrm{x}]\{\mathrm{a}\} = \{\mathrm{y}\} \tag{3}$$

In order to perform linear regression in Matlab the objective is to determine the vector “{a}” from Eq. (3). This is done by using the formula given below:

$$\{a\} = [x] \backslash \{y\} \tag{4}$$

Notice that what is called matrix [x] was built by combining each of the individual independent variable column vectors $\{x_1\}$, $\{x_2\}$, $\{x_3\}$ and a unit column vector (vector which constituents are all 1), as shown in the schematic representation given below:

$$\begin{array}{ccccccccc}
\boxed{a_1} x_{1,1} & + & \boxed{a_2} x_{2,1} & + & \boxed{a_3} x_{3,1} & +.... & \boxed{a_m} & = & y_1 \\
a_1 x_{1,2} & + & a_2 x_{2,2} & + & a_3 x_{3,2} & +.... & a_m & = & y_2 \\
a_1 x_{1,3} & + & a_2 x_{2,3} & + & a_3 x_{3,3} & +.... & a_m & = & y_3 \\
 & & & & : & & & & \\
a_1 x_{1,n} & + & a_2 x_{2,n} & + & a_3 x_{3,n} & +.... & a_m & = & y_n
\end{array}$$

$$\begin{bmatrix}
x_{1,1} & x_{2,1} & x_{3,1} & \dots & 1 \\
x_{1,2} & x_{2,2} & x_{3,2} & \dots & 1 \\
x_{1,3} & x_{2,3} & x_{3,3} & \dots & 1 \\
 & : & & \dots & 1 \\
x_{1,n} & x_{2,n} & x_{3,n} & \dots & 1
\end{bmatrix}
\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ : \\ a_m \end{bmatrix}
=
\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ : \\ y_n \end{bmatrix}$$

Unit vector

The procedure to perform linear regression in Matlab is summarized below:

1. Input the experimental data in the mfile. For example, input the vectors {y}, {x_1}, {x_2}, {x_3} etc. Make sure that the vectors are column vectors. If you input the vectors as row vectors use the transpose (See Tutorial III, p.6)
2. Create the unit column vector
3. Create the matrix [x] by combining each of the individual column vectors and the unit vector (See Tutorial III, p. 4)
4. Apply Eq. (4) to calculate the coefficient vector {a}. These are the parameters for your equation.
5. Determine how good is your fit by:
   a. Calculate the predicted value
   b. Calculate the difference between the predicted value and the experimental value
   c. Make a table that shows the differences (experimental data, predicted value, and difference between experimental data and predicted value)
   d. Plot the experimental data (using plot see Tutorial V.b)
   e. Plot the equation (using fplot see Tutorial V.b)

Solved Problem using Matlab:

Develop a linear correlation to predict the final weight of an animal based on the initial weight and the amount of feed eaten.

$$(\text{final weight}) = a_1(\text{initial weight}) + a_2*(\text{feed eaten}) + a_3$$

The following data are given:

final weight (lb)	initial weight (lb)	feed eaten (lb)
95	42	272
77	33	226
80	33	259
100	45	292
97	39	311
70	36	183
50	32	173
80	41	236
92	40	230
84	38	235

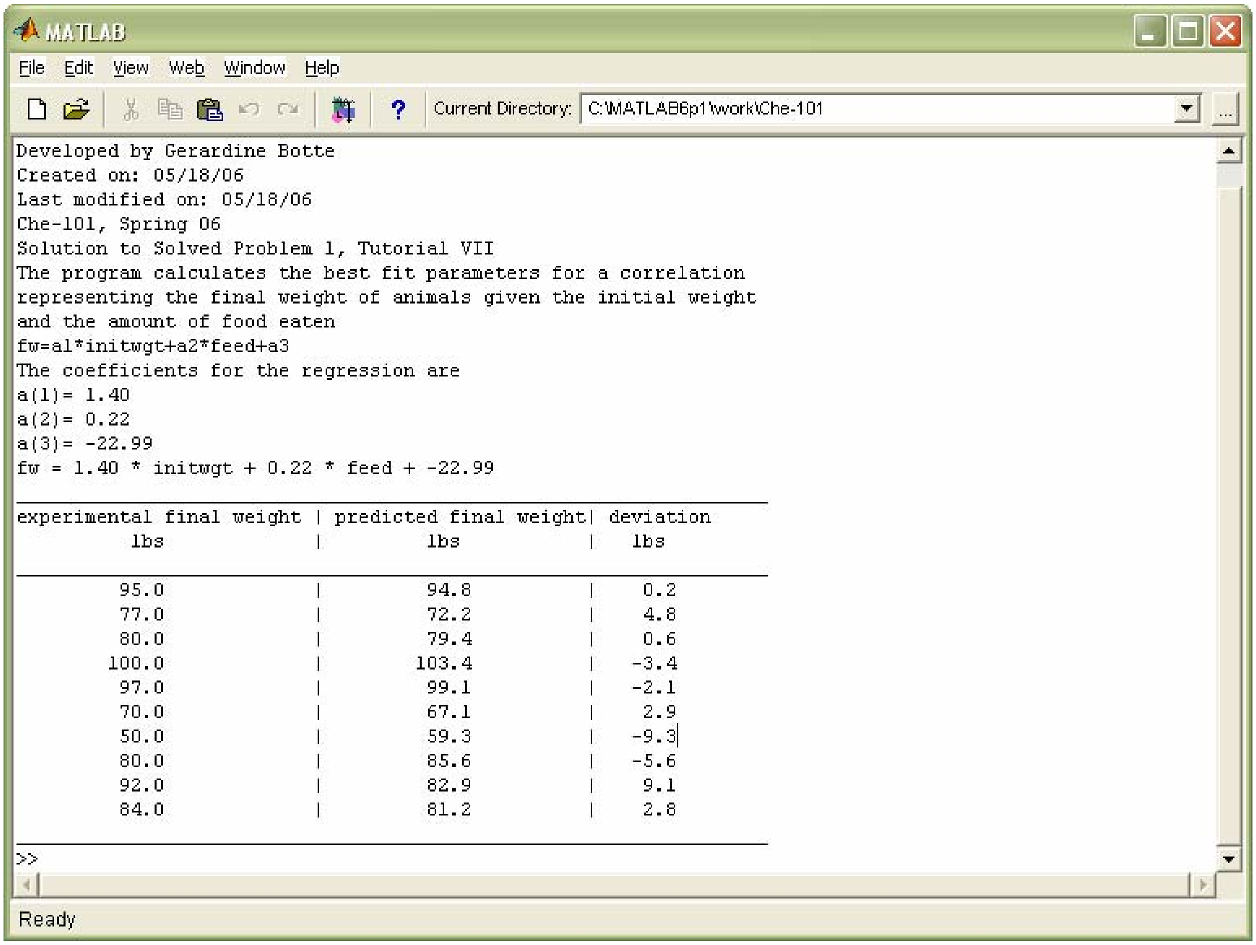
Solution:

The mfile is shown below:

```
% This program shows an example of linear regression in Matlab
% Developed by Gerardine Botte
% Created on: 05/18/06
% Last modified on: 05/18/06
% Che-101, Spring 06
% Solution to Solved Problem 1, Tutorial VII
% The program calculates the best fit parameters for a correlation
% representing the final weight of animals given the initial weight
% and the amount of food eaten:
%  fw=a1*initwgt+a2*feed+a3
%--------------------------------

clear;
clc;

fprintf('This program shows an example of linear regression in Matlab\n');
fprintf('Developed by Gerardine Botte\n');
fprintf('Created on: 05/18/06\n');
fprintf('Last modified on: 05/18/06\n');
fprintf('Che-101, Spring 06\n');
fprintf('Solution to Solved Problem 1, Tutorial VII\n');
fprintf('The program calculates the best fit parameters for a correlation\n');
fprintf('representing the final weight of animals given the initial weight\n');
fprintf('and the amount of food eaten\n');
fprintf('fw=a1*initwgt+a2*feed+a3\n');


%Step 1 of Procedure (see p. 3, TVII): input the data into vectors.
initwgt = [ 42 33 33 45 39 36 32 41 40 38];        % in lbs. (independent variable)
feed = [ 272 226 259 292 311 183 173 236 230 235];  % in lbs. (independent variable)
fw = [95; 77; 80; 100; 97; 70; 50; 80; 92; 84];     % in lbs (dependent variable).

%because the data is given as row vectors it needs to be transformed into column vectors
initwgt=initwgt';
feed=feed';

%Step 2 of Procedure (see p. 3, TVII): Create the unit column vector
for i=1:10
unit(i)=1;
end
unit=unit';

%Step 3 of Procedure (see p.3, TVII):Create the matrix [x] by combining each of
%     the individual column vectors and the unit vector (See Tutorial III, p. 4)

x=[initwgt feed unit];

%Step 3 of Procedure (see p.3, TVII):4.  Apply Eq. (4) to calculate the coefficient
%                 vector {a}. These are the parameters for your equation.

a=x\fw;

%Make sure to bring all the vectors back into row vectors so that you can use for loops
%for printingand performing vector operations

%printing the parameters
initwgt=initwgt';
feed=feed';
a=a';
fw=fw';

fprintf('The coefficients for the regression are\n');
for i=1:3
    fprintf('a(%1i)= %4.2f\n', i, a(i));
end

%you can also print the equation by using fprintf:
fprintf('fw = %4.2f * initwgt + %4.2f * feed + %4.2f\n', a(1), a(2), a(3));

%Calculating the numbers predicted by the equation and the difference
for i=1:10
    fwp(i)=initwgt(i)*a(1)+feed(i)*a(2)+a(3); %This is the predicted final weight, lbs
    dev(i)=fw(i)-fwp(i); %this is the deviation, lbs
end

%Making the comparison table:
```

```
fprintf('______________________________________________________________\n');
fprintf('experimental final weight | predicted final weight| deviation\n');
fprintf('         lbs              |          lbs          |  lbs\n');
fprintf('______________________________________________________________\n');
for i=1:10
   fprintf('       %5.1f            |        %5.1f         | %5.1f\n', fw(i), fwp(i), dev(i));
end
fprintf('______________________________________________________________\n');
```

This is what you will see on screen:

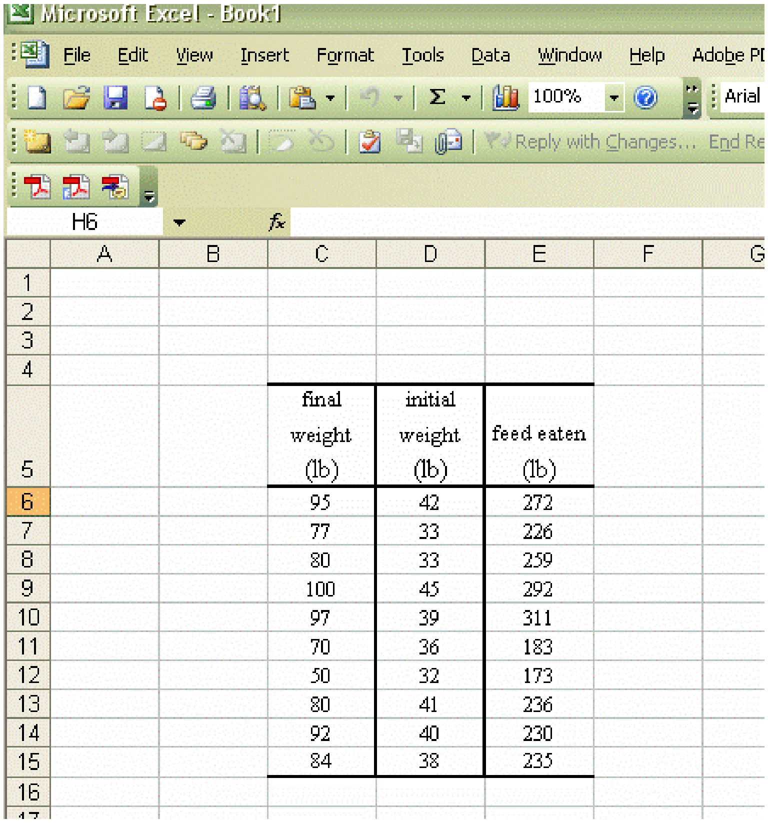

MATLAB
File Edit View Web Window Help
Current Directory: C:\MATLAB6p1\work\Che-101
Ready

```
Developed by Gerardine Botte
Created on: 05/18/06
Last modified on: 05/18/06
Che-101, Spring 06
Solution to Solved Problem 1, Tutorial VII
The program calculates the best fit parameters for a correlation
representing the final weight of animals given the initial weight
and the amount of food eaten
fw=a1*initwgt+a2*feed+a3
The coefficients for the regression are
a(1)= 1.40
a(2)= 0.22
a(3)= -22.99
fw = 1.40 * initwgt + 0.22 * feed + -22.99
_________________________________________________________________
experimental final weight | predicted final weight| deviation
          lbs             |          lbs          |   lbs
_________________________________________________________________
         95.0             |         94.8          |    0.2
         77.0             |         72.2          |    4.8
         80.0             |         79.4          |    0.6
        100.0             |        103.4          |   -3.4
         97.0             |         99.1          |   -2.1
         70.0             |         67.1          |    2.9
         50.0             |         59.3          |   -9.3
         80.0             |         85.6          |   -5.6
         92.0             |         82.9          |    9.1
         84.0             |         81.2          |    2.8
_________________________________________________________________
>>
```

Procedure to perform linear regression in Excel:

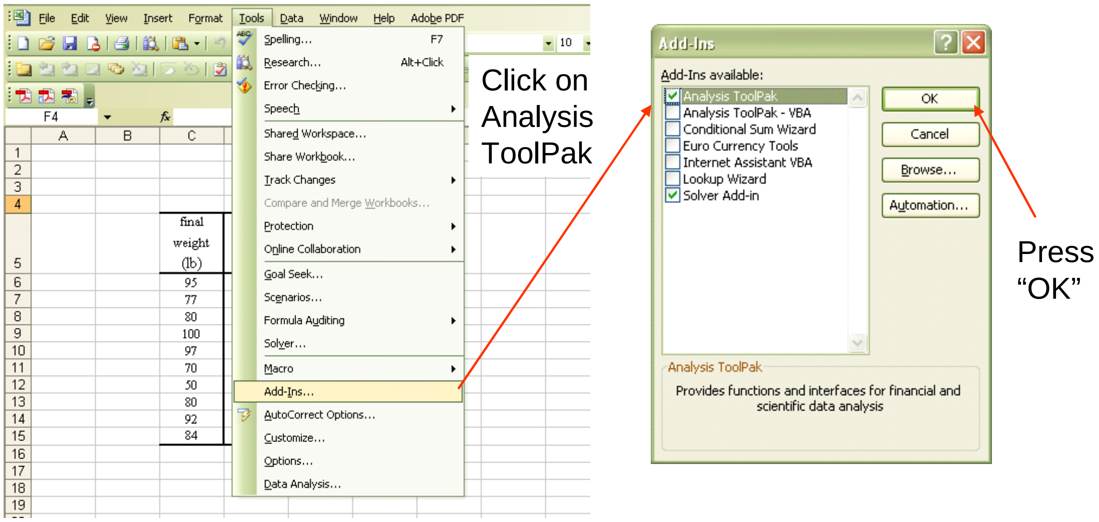
Excel can do single or multiple linear regression through the “data analysis” toolbox. This toolbox needs to be added as an “add in”. To illustrate how to perform linear regression in Excel let us solve the same problem:

1. Write your data into an Excel spreadsheet as shown below:


Microsoft Excel - Book1
File Edit View Insert Format Tools Data Window Help Adobe PD
100%
Arial
Reply with Changes... End Re
H6
fx

final weight (lb)	initial weight (lb)	feed eaten (lb)
95	42	272
77	33	226
80	33	259
100	45	292
97	39	311
70	36	183
50	32	173
80	41	236
92	40	230
84	38	235

2. Load the "data analysis toolbox" :



Click on Analysis ToolPak

Add-Ins

Add-Ins available:

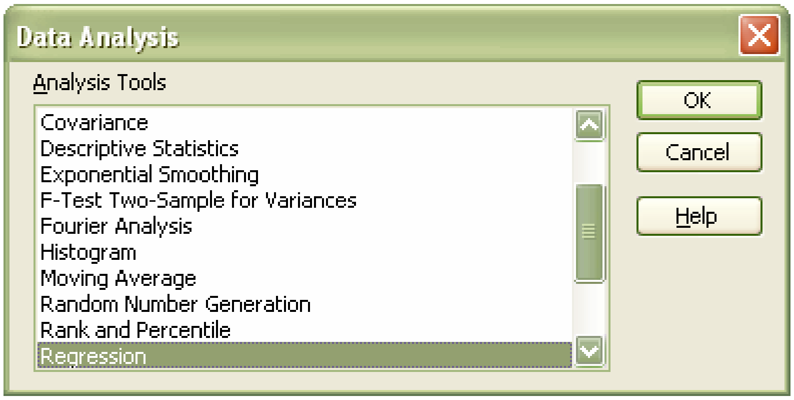
☑ Analysis ToolPak
☐ Analysis ToolPak - VBA
☐ Conditional Sum Wizard
☐ Euro Currency Tools
☐ Internet Assistant VBA
☐ Lookup Wizard
☑ Solver Add-in

Analysis ToolPak

Provides functions and interfaces for financial and scientific data analysis

Press "OK"

3. Go to “Data Analysis” and find the “Regression” tool:



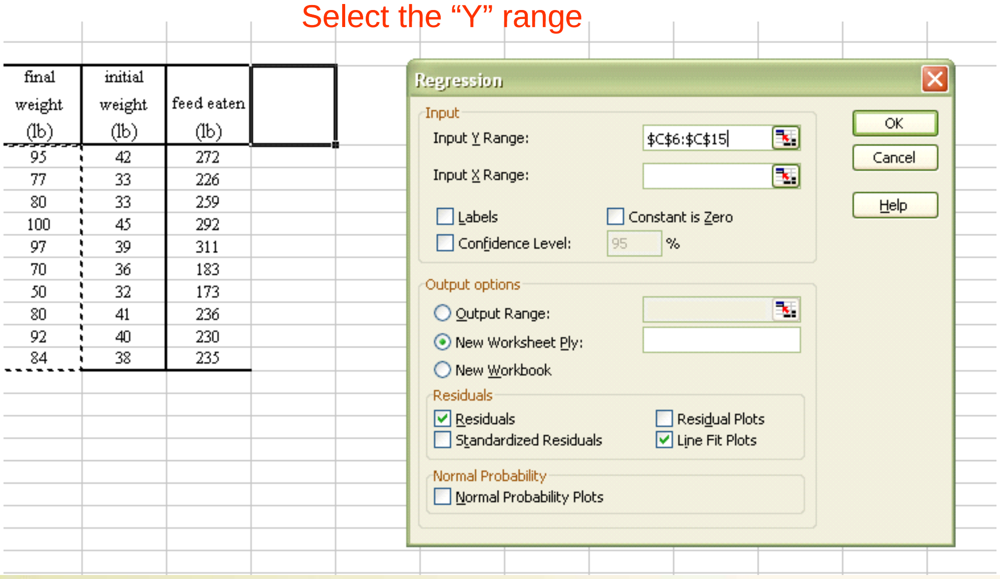
Data Analysis

Analysis Tools

- Covariance
- Descriptive Statistics
- Exponential Smoothing
- F-Test Two-Sample for Variances
- Fourier Analysis
- Histogram
- Moving Average
- Random Number Generation
- Rank and Percentile
- Regression

4. Click “OK” and you will be prompted to the Regression analysis:

Select the “Y” range



final weight (lb)	initial weight (lb)	feed eaten (lb)
95	42	272
77	33	226
80	33	259
100	45	292
97	39	311
70	36	183
50	32	173
80	41	236
92	40	230
84	38	235

Regression

Input

Input Y Range: $C$6:$C$15

Input X Range:

□ Labels □ Constant is Zero

□ Confidence Level: 95 %

Output options

○ Output Range:

● New Worksheet Ply:

○ New Workbook

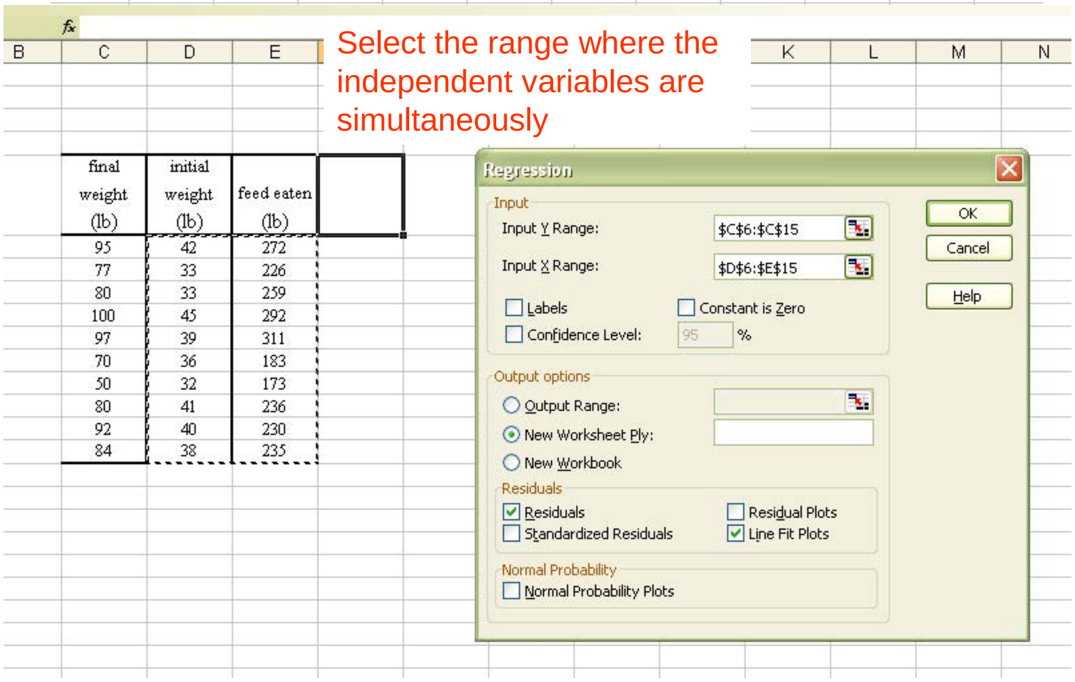
Residuals

☑ Residuals □ Residual Plots

□ Standardized Residuals ☑ Line Fit Plots

Normal Probability

□ Normal Probability Plots



Select the range where the independent variables are simultaneously

final weight (lb)	initial weight (lb)	feed eaten (lb)
95	42	272
77	33	226
80	33	259
100	45	292
97	39	311
70	36	183
50	32	173
80	41	236
92	40	230
84	38	235

Regression

Input

Input Y Range: $C$6:$C$15

Input X Range: $D$6:$E$15

□ Labels □ Constant is Zero

□ Confidence Level: 95 %

Output options

○ Output Range:

● New Worksheet Ply:

○ New Workbook

Residuals

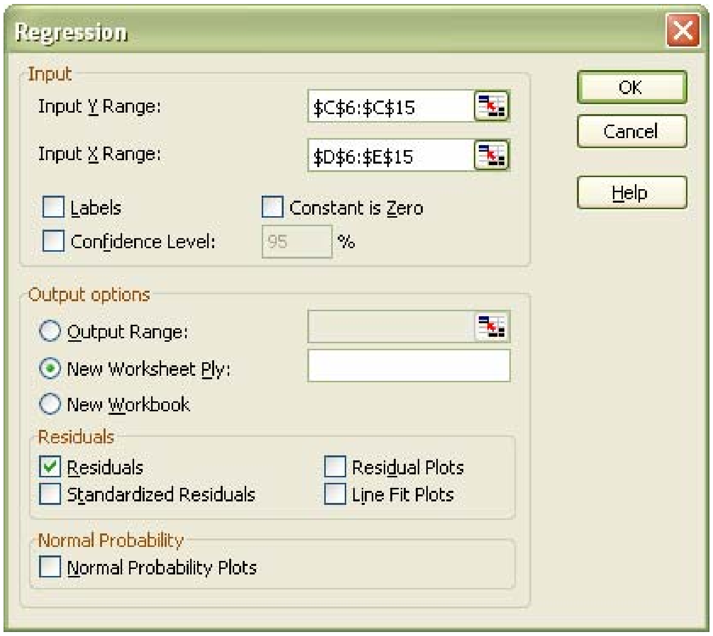
☑ Residuals □ Residual Plots

□ Standardized Residuals ☑ Line Fit Plots

Normal Probability

□ Normal Probability Plots

5. Make the additional selections and press “OK”



Input

Input Y Range: $C$6:$C$15

Input X Range: $D$6:$E$15

□ Labels □ Constant is Zero

□ Confidence Level: 95 %

Output options

○ Output Range:

● New Worksheet Ply:

○ New Workbook

Residuals

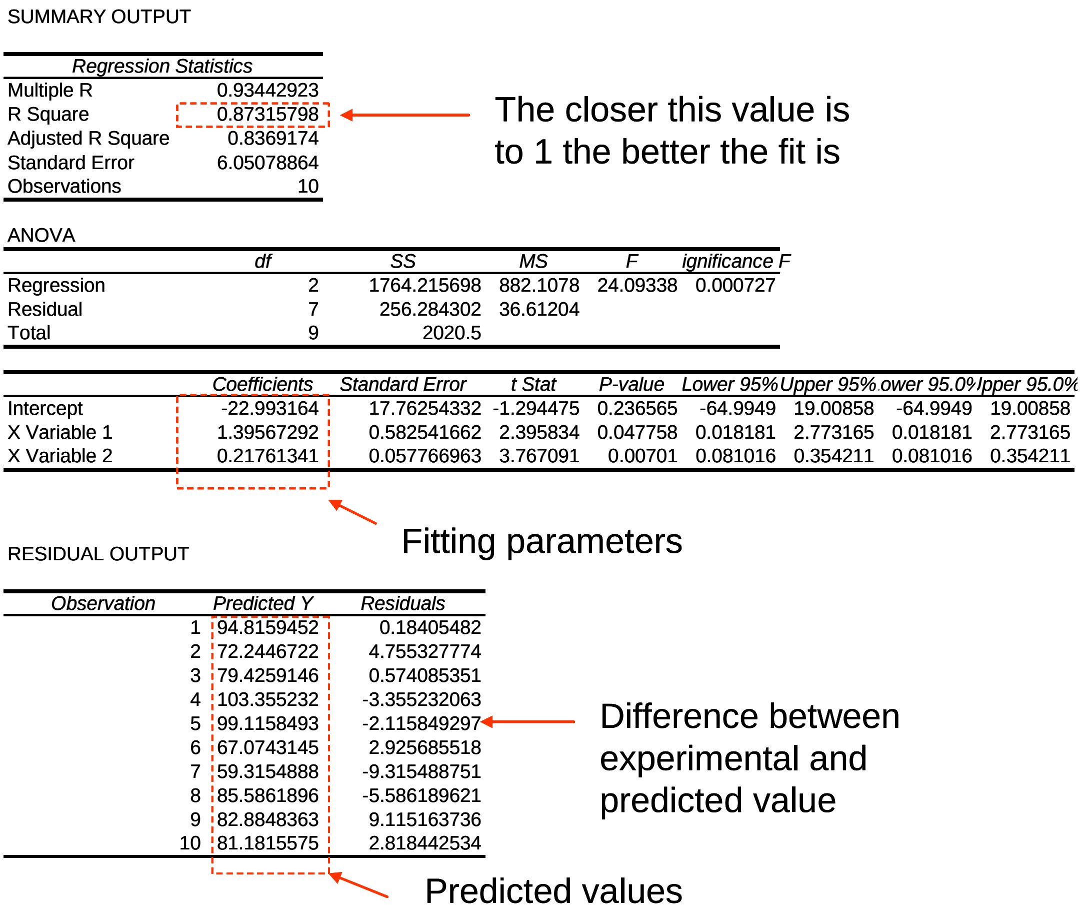
☑ Residuals □ Residual Plots

□ Standardized Residuals □ Line Fit Plots

Normal Probability

□ Normal Probability Plots

6. This is what you will see on the screen:



SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.93442923
R Square	0.87315798
Adjusted R Square	0.8369174
Standard Error	6.05078864
Observations	10

The closer this value is to 1 the better the fit is

ANOVA

	df	*SS*	*MS*	*F*	*ignificance F*
Regression	2	1764.215698	882.1078	24.09338	0.000727
Residual	7	256.284302	36.61204		
Total	9	2020.5			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*	*ower 95.0%*	*Jpper 95.0%*
Intercept	-22.993164	17.76254332	-1.294475	0.236565	-64.9949	19.00858	-64.9949	19.00858
X Variable 1	1.39567292	0.582541662	2.395834	0.047758	0.018181	2.773165	0.018181	2.773165
X Variable 2	0.21761341	0.057766963	3.767091	0.00701	0.081016	0.354211	0.081016	0.354211

Fitting parameters

RESIDUAL OUTPUT

Observation	*Predicted Y*	*Residuals*
1	94.8159452	0.18405482
2	72.2446722	4.755327774
3	79.4259146	0.574085351
4	103.355232	-3.355232063
5	99.1158493	-2.115849297
6	67.0743145	2.925685518
7	59.3154888	-9.315488751
8	85.5861896	-5.586189621
9	82.8848363	9.115163736
10	81.1815575	2.818442534

Difference between experimental and predicted value

Predicted values

7. You will learn how to interpret more of the statistical results in the Experimental Design Course.