بسم الله الرحمن الرحيم

DEPARTMENT OF CIVIL ENGINEERING – KFUPM

Numerical and Statistical Methods in Civil Engineering

CE 318- 51- 2011-2012 (111)

Computer Lab. Sessions NO. 03 & 04

Subj.: Using *Mathematica* software for Curve-fitting.

DATE: Oct. 03,'11

Objective: Basic introduction to *Mathematica software* mainly to perform function-graph and function-curve *linear* and *nonlinear* fitting using *the software*.

Plotting Lab Data using *Mathematica software*©

The purpose of these notes is to show how *Mathematica* can be used to plot and analyze laboratory data. This is a shortened version of the document, *Graph Plotting and Data Analysis using Mathematica*, which you should read for a more complete description of *Mathematica* plotting commands.

It is a good practice to frequently **reset** everything before you begin a *Mathematica* session.

```
In[1]:= Clear["Global` * "]
```

Reading data from a laboratory experiment:

Most of the data that you obtain in the laboratory will consists of pairs of (x,y) values, for *example*:

```
In[2]:= data ={{0 ,6.62},{1 ,6.73},{2 ,6.86}, {3 ,6.98},{4 ,7.03}};
```

One problem with this method of data entry is that becomes laborious to type many curly brackets and commas, as well as increasing the possibility of making mistakes. An alternate method is to *first create a data file* using a text editor. *A file which consists of two columns of data might look like this*, with a *space between two columns* of numbers:

```
 8.1   .5
 9.2   1
10.5   1.5
13.1   2
15.4   2.5
 18    3
20.4   3.5
22.9   4
24.5   4.5
26.3   5
```

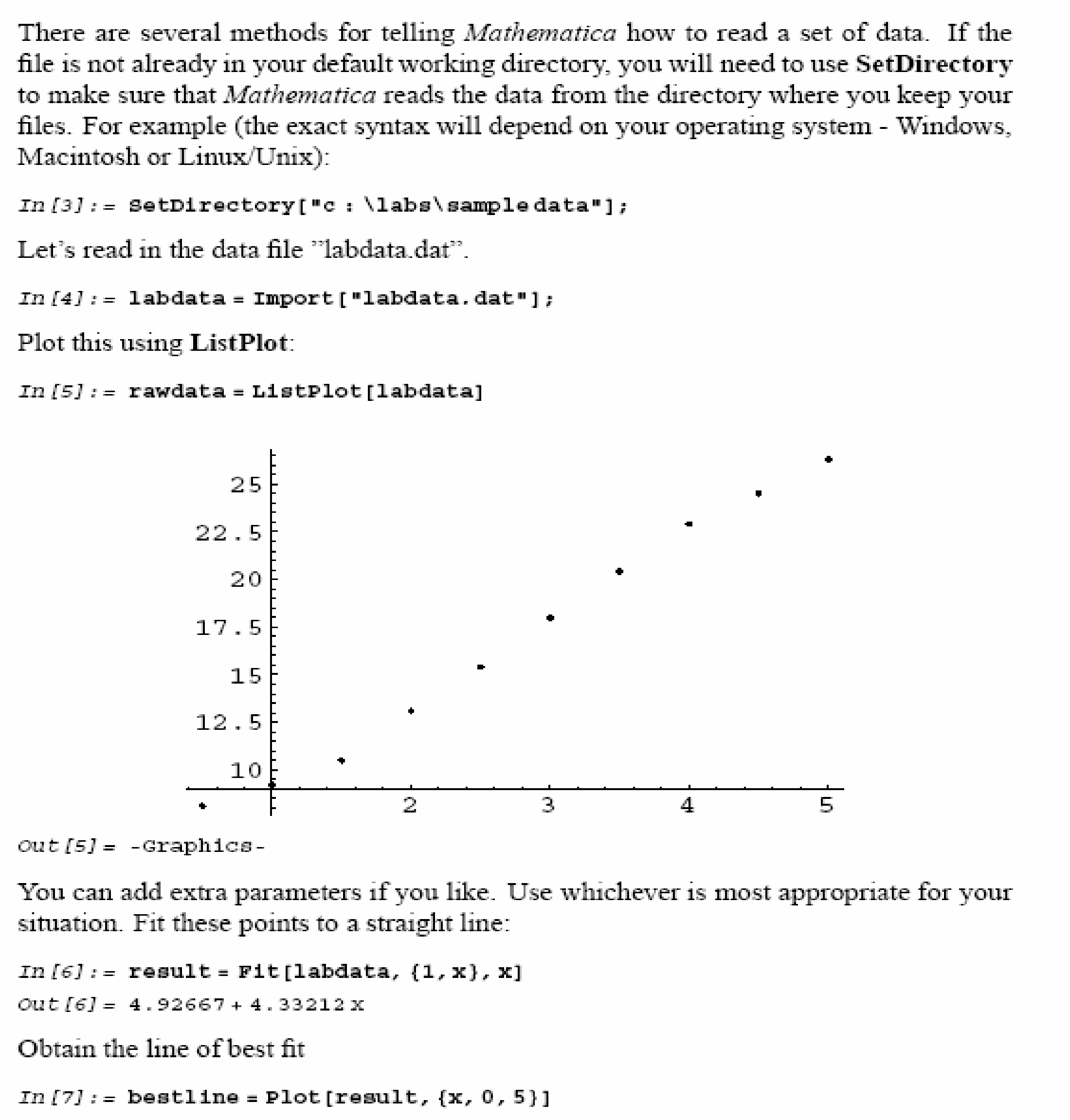
Save the file under a meaningful name, such as "**labdata.dat**". Then use *Mathematica* commands to refer to the **data-file** to *plot* the data and *perform function curve fitting*.

Utilize the notes provided (herewith) to run the sample cases provided and then do the same to obtain a ***quadratic*** *and a* ***cubic curve-fitting*** *for the following data.* Comapre your results.

x : 22.3, 27.3, 29.7, 33.2, 39.7, 44.7, 49.6, 62.1, 67, 74.5, 84.4, 94.9, 99.3

y : 1501, 1298, 1054, 987, 905, 824, 643, 581, 555, 505, 398, 344, 327

Basic Selected Notes on *Mathematica*:



There are several methods for telling *Mathematica* how to read a set of data. If the file is not already in your default working directory, you will need to use **SetDirectory** to make sure that *Mathematica* reads the data from the directory where you keep your files. For example (the exact syntax will depend on your operating system - Windows, Macintosh or Linux/Unix):

```
In[3]:= SetDirectory["c:\labs\sampledata"];
```

Let's read in the data file "labdata.dat".

```
In[4]:= labdata = Import["labdata.dat"];
```

Plot this using **ListPlot**:

```
In[5]:= rawdata = ListPlot[labdata]
```

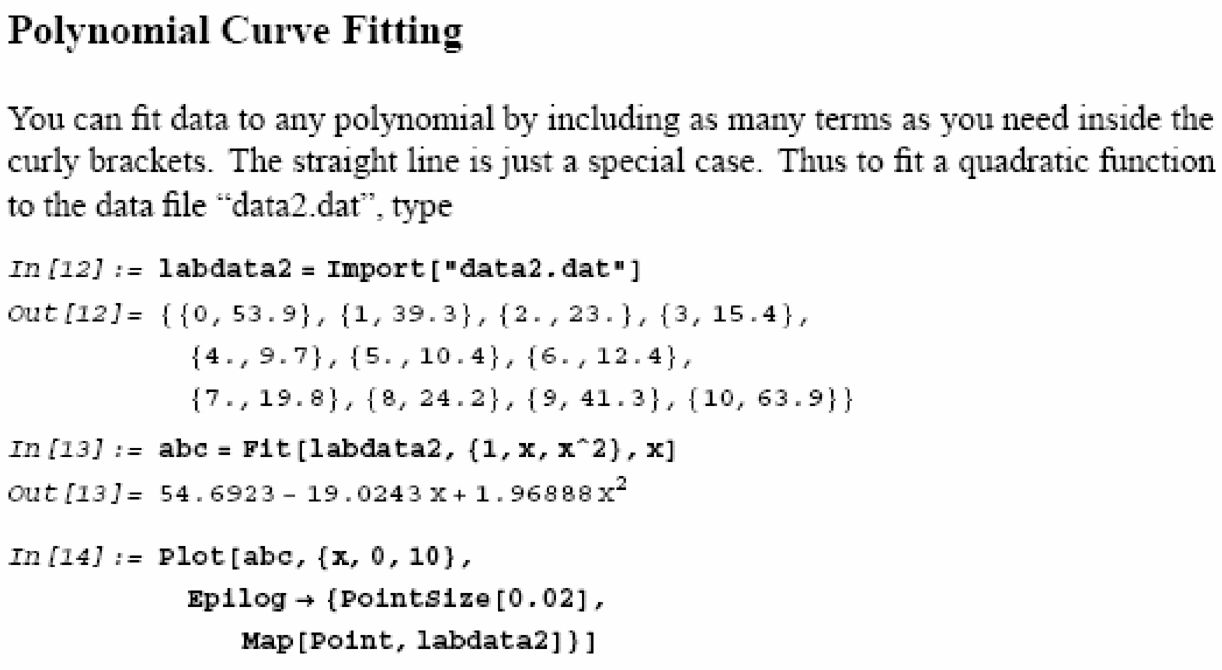
```
Out[5]= -Graphics-
```

You can add extra parameters if you like. Use whichever is most appropriate for your situation. Fit these points to a straight line:

```
In[6]:= result = Fit[labdata, {1, x}, x]
Out[6]= 4.92667 + 4.33212 x
```

Obtain the line of best fit

```
In[7]:= bestline = Plot[result, {x, 0, 5}]
```



Polynomial Curve Fitting

You can fit data to any polynomial by including as many terms as you need inside the curly brackets. The straight line is just a special case. Thus to fit a quadratic function to the data file "data2.dat", type

```
In[12]:= labdata2 = Import["data2.dat"]
Out[12]= {{0, 53.9}, {1, 39.3}, {2., 23.}, {3, 15.4},
           {4., 9.7}, {5., 10.4}, {6., 12.4},
           {7., 19.8}, {8, 24.2}, {9, 41.3}, {10, 63.9}}
In[13]:= abc = Fit[labdata2, {1, x, x^2}, x]
Out[13]= 54.6923 - 19.0243 x + 1.96888 x^2

In[14]:= Plot[abc, {x, 0, 10},
           Epilog → {PointSize[0.02],
              Map[Point, labdata2]}]
```

Nonlinear Curve Fitting

If your data does not follow a straight line or simple polynomial, you will need to use *Mathematica*'s NonlinearFit functions:

```
In[21]:= << Statistics`NonlinearFit`
```

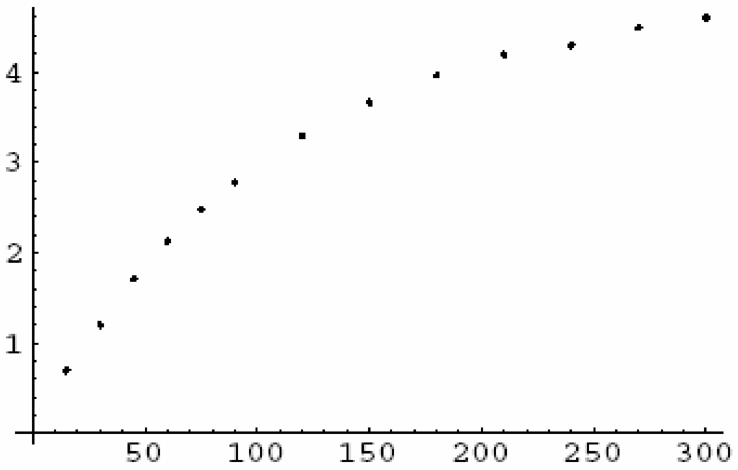
Example: Charging a Capacitor

In this experiment we measure the voltage across the capacitor as a function of time. As usual, we'll read the data from a file:

```
In[22]:= chargedata = Import["capch.dat"]
Out[22]= {{15, 0.7}, {30, 1.2}, {45, 1.71}, {60, 2.13},
           {75, 2.48}, {90, 2.78}, {120, 3.29},
           {150, 3.66}, {180, 3.96}, {210, 4.19},
           {240, 4.29}, {270, 4.49}, {300, 4.6}}
```

Plot in the usual way:

```
In[23]:= cdata = ListPlot[chargedata]
```



```
Out[23]= -Graphics-
```

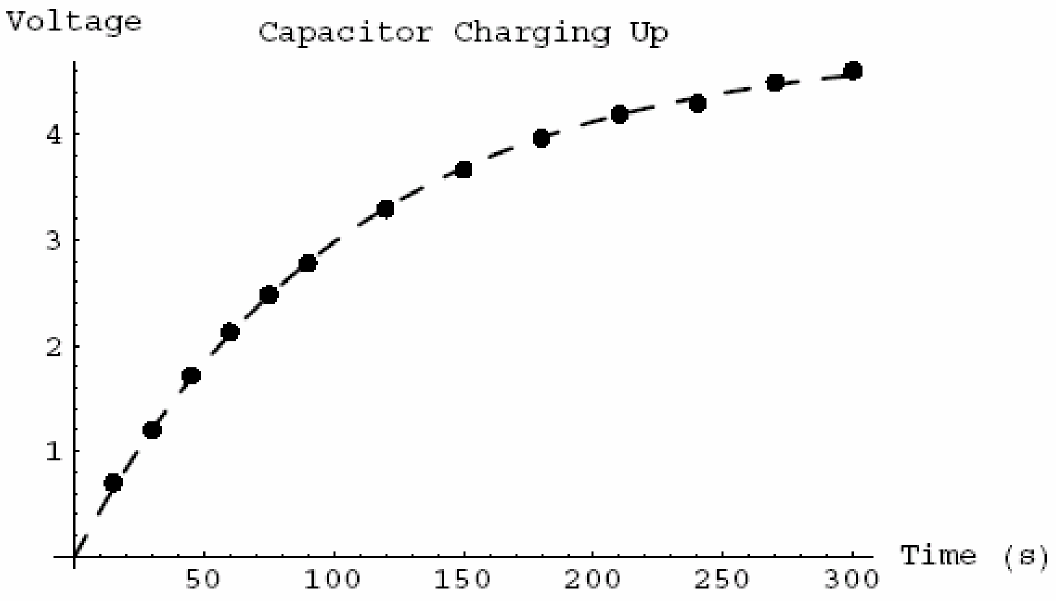
Define the function and ask Mathematica to solve for a and b:

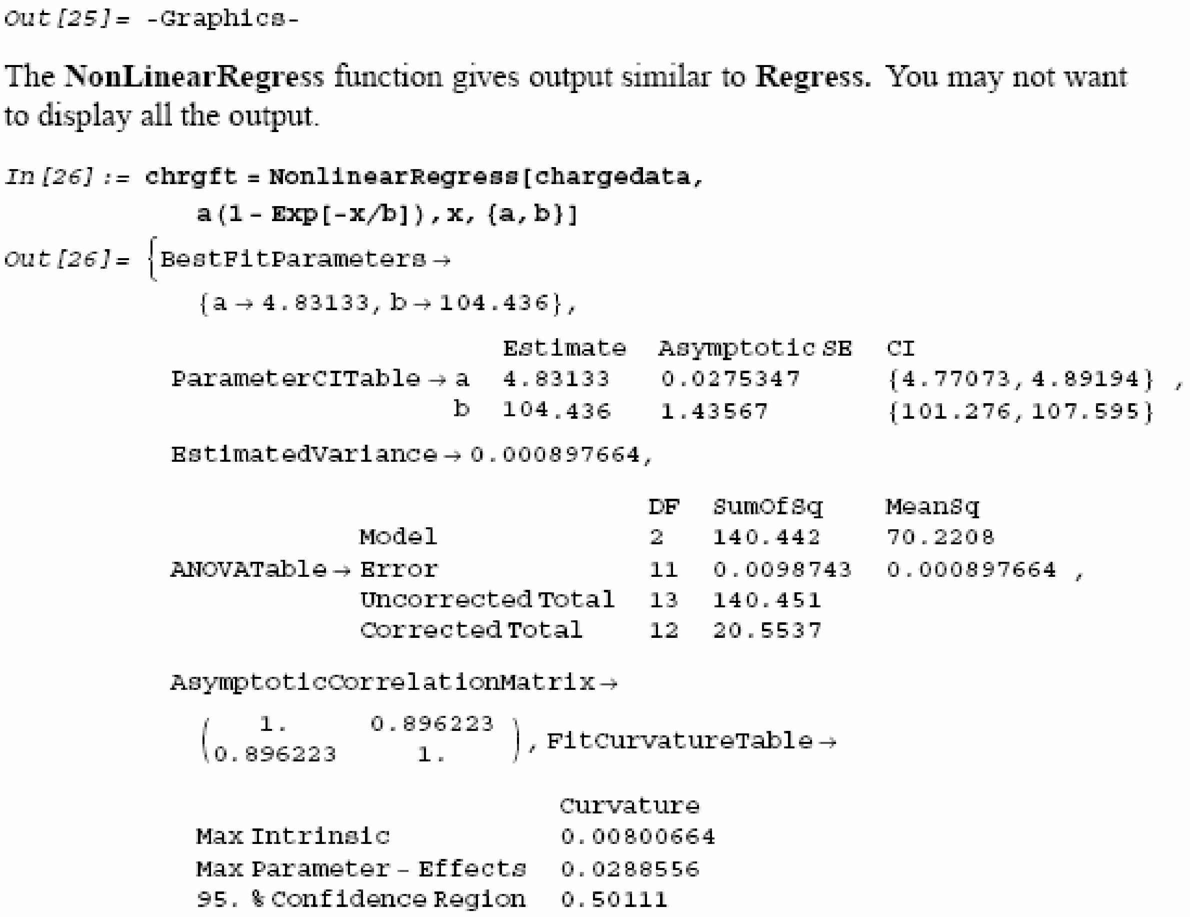
```
In[24]:= chrgft = NonlinearFit[chargedata,
             a(1 - Exp[-x/b]), x, {a, b}]
```

Out[24]= $4.83133\,(1 - e^{-0.00957528\,x})$

Plot it, adding a few extra features:

```
In[25]:= Plot[chrgft, {x, 0, 300},
           AxesLabel- > {"Time (s)", "Voltage"},
           PlotLabel → "Capacitor Charging Up",
           PlotStyle →
             {{Dashing[{0.03}], Thickness[0.005]}},
           Epilog → {PointSize[0.02],
               Map[Point, chargedata]}]
```





```
Out[25]= -Graphics-
```

The **NonLinearRegress** function gives output similar to **Regress.** You may not want to display all the output.

```
In[26] := chrgft = NonlinearRegress[chargedata,
                a(1 - Exp[-x/b]), x, {a, b}]
Out[26]= {BestFitParameters →
                {a → 4.83133, b → 104.436},

                                   Estimate   Asymptotic SE   CI
          ParameterCITable → a     4.83133    0.0275347       {4.77073, 4.89194} ,
                             b     104.436    1.43567         {101.276, 107.595}

          EstimatedVariance → 0.000897664,

                                                DF   SumOfSq     MeanSq
                        Model                   2    140.442     70.2208
          ANOVATable →  Error                   11   0.0098743   0.000897664 ,
                        Uncorrected Total       13   140.451
                        Corrected Total         12   20.5537

          AsymptoticCorrelationMatrix →

           (   1.       0.896223 )
           ( 0.896223     1.     ), FitCurvatureTable →

                                        Curvature
           Max Intrinsic                0.00800664
           Max Parameter - Effects      0.0288556
           95. % Confidence Region      0.50111
```

****

Procedure for Lab.-Report Evaluation:

1. Start working in the assigned session, then complete your computer works *preferably* within the session *or* shortly afterwards using the same computing machine on which you may save your work for future use (if necessary).
2. Submit for evaluation your summary of organized computer work assignment in the beginning of the next lab.
3. Your report *should* include: i)**Introduction** expalining the work undertaken and its main objectives; ii) Clear outline of the numerical procedure(s) used; iii) **Print-out** of the work completed; and iv) **Summary** and conclusions.