

Nonlinear Least Squares Regression ©

by

Sidney H. Young and Andrzej Wierzbicki

Department of Chemistry
University of South Alabama
Mobile, AL. 36688

syoung@jaguar1.usouthal.edu

© Copyright 2000 by the Division of Chemical Education, Inc., American Chemical Society. All rights reserved. For classroom use by teachers, one copy per student in the class may be made free of charge. Write to JCE Online, jceonline@chem.wisc.edu, for permission to place a document, free of charge, on a class Intranet.

Introduction

Nonlinear least squares regression is often required in the Physical Chemistry Laboratory. This template demonstrates various implicit and explicit methods for determination of the parameters of the regressed curve. It will produce standard deviation of fit, and standard deviations of the parameters. Residual analysis is used to demonstrate techniques of removing bad data points from the fit. This template may read in data from a file, allowing it to be used in general. You may need to adjust the read statements in this document to match the version of Mathcad that you are using.

As in the accompanying template, Linear Least Squares Regression, the parameters will be tested to see if their addition in the model is statistically significant.

Goal

To teach the techniques of nonlinear least squares regression, and to be a practical aid in the Physical Chemistry Laboratory.

Prerequisites

1. Moderate skill with Mathcad in performing simple calculations and preparing plots.
2. Familiarity with the basic concepts of nonlinear regression.

Performance Objectives

At the end of the exercise, you will be able to:

1. determine the parameters of a regression line, given x-y data pairs;
2. find the standard deviation of fit and standard deviation of parameters;
3. use residual analysis to look for bad data points;
4. test if the addition of an extra parameter is statistically significant with respect to the quality of the fit.

I. Test Data

The fluorescence spectra of myoglobin as a function of added guanidine hydrochloride was measured to determine the effect of the latter on protein folding. The fluorescence peak at 340 nm was monitored with an excitation wavelength of 285 nm. See C. Jones, *J. Chem. Ed.*, Vol. 74, No. 11, 1306-1310. (1997).

$$I := \begin{bmatrix} 7237 \\ 11513 \\ 13419 \\ 15031 \\ 21498 \\ 51848 \\ 71653 \end{bmatrix} \quad C := \begin{bmatrix} 0 \\ 0.806 \\ 0.968 \\ 1.21 \\ 1.61 \\ 2.0 \\ 4.836 \end{bmatrix} \quad \begin{array}{l} I_n := 7237 \\ I_u := 71653 \end{array} \quad \begin{array}{l} \text{Fluorescence in native} \\ \text{state} \\ \text{Fluorescence when completely} \\ \text{unfolded.} \end{array}$$

I is the intensity of myoglobin fluorescence;
C is the molarity of the guanidine hydrochloride solution.

Exercise 1: Determine the limits for fluorescence intensity I and folding fraction f at zero and infinite concentration.

Preparing a file of data for Mathcad:

For experimental data set, generate a 2-column data array using your favorite software. Place the the dependent variable in column 1 and the independent variable in column 2. Save the file as a comma or tab delimited ASCII file and call this file `filenam.prn`. Then use the following commands:

```
Z:=READPRN("filenam.prn")  
I:Z Ctrl-Shift-6 0  
C:Z Ctrl-Shift-6 1
```

```
N:length(I)  
In:I[0  
Iu:I[N-1
```

The above commands set up I and C from the input data, define N as the number of data points, and define In, the intensity of the native state, and Iu, the completely unfolded state.

Note that the first point (C=0) corresponds to the protein in its native state and last point (C=4.836 M guanidine hydrochloride) corresponds to the fully unfolded state. In general, $I_n := I_0$ and $I_u := I_{N-1}$ where N is the number of data points

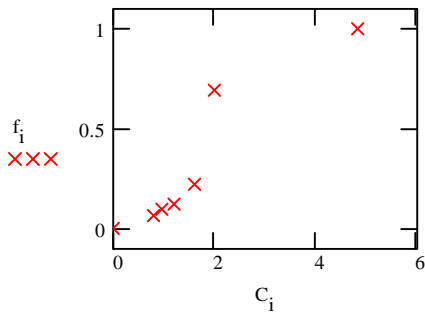
$N := \text{length}(I)$ $N = 7$ Number of data points in data arrays I and C
 $i := 0.. N - 1$

Now we define the folding fraction. Imagine that there are two states of a protein, folded and unfolded. At any concentration of guanidine hydrochloride the fraction of the protein concentration which is in the unfolded state relative to the total amount of protein is the folding fraction. Assuming that the fluorescence intensity is linearly proportional to the concentration of each state (with different proportionalities), then the folding fraction is defined using linear interpolation. In other words, since $I_n - I_u$ is proportional to the amount of unfolded protein and $I_n - I_i$ is proportional to the amount in the unfolded state at concentration C_i , it follows that f_i is the fraction in an unfolded state.

The arrow over the term $I_n - I_i$ is the vectorize symbol, allowing f_i to be calculated for each value of i simultaneously.

$$f_i := \frac{\overrightarrow{I_n - I_i}}{I_n - I_u}$$

Folding fraction is the amount unfolded protein divided by the total amount of protein.



Plot of folding fraction vs. guanidine hydrochloride concentration. How does the folding fraction behave as the concentration of guanidine hydrochloride increases?

It is clear from the above graph that a linear plot is unsatisfactory. Actually, the folding fraction as a function of guanidine concentration is given by the following mathematical model:

Developing the mathematical model: Thermodynamics provides the tools for us to model the folding fraction. First we assume that equilibrium exists between folded and unfolded moieties, and that the denaturant guanidine hydrochloride gives rise to a linear perturbation of the Gibbs free energy. This is shown in the following equation.

$$\Delta G_U = \Delta G_{\text{water}} - \text{rate} \cdot C$$

where ΔG_U is the free energy of unfolding, ΔG_{water} is the free energy in the absence of guanidine hydrochloride, **rate** is the rate of change of free energy as a function of guanidine concentration, and C is the concentration of guanidine hydrochloride.

At the point at which half of the protein is unfolded, the free energy is zero (because the equilibrium constant is unity) and thus

$$\Delta G_{\text{water}} = \text{rate} \cdot C_m$$

where C_m is the concentration of guanidine hydrochloride at the half-unfolded state.

$$\text{Thus } \Delta G_U = \text{rate} \cdot (C_m - C) .$$

Now we use $K = e^{\frac{-\Delta G_U}{R \cdot T}}$ to obtain the equilibrium constant where $K = \frac{U}{N}$,

U is the concentration of unfolded protein, and N is the concentration of folded protein. Finally, the folding fraction f is $\frac{U}{N + U}$. Dividing the numerator and denominator of f (see below) by N and insert K . This gives

$$f = \frac{\left(\frac{U}{N}\right)}{1 + \left(\frac{U}{N}\right)} = \frac{K}{1 + K} \quad \text{which finally becomes}$$

$$f = \frac{e^{\frac{-\Delta G_U}{R \cdot T}}}{1 + e^{\frac{-\Delta G_U}{R \cdot T}}} \quad \text{when the definition of the equilibrium constant } K \text{ is used.}$$

$$f = \frac{e^{-\frac{\text{rate} \cdot (C_m - C)}{R \cdot T}}}{1 + e^{-\frac{\text{rate} \cdot (C_m - C)}{R \cdot T}}}$$

The equation we need for our data appears after we substitute the definition for ΔG_U , for our equilibrium.

Note that there are two parameters, **rate** and **C_m**. The variables are **C**, the concentration of guanidine hydrochloride, and **f**, the folding fraction, determined from fluorescence intensities. To review, **rate** is the is the rate of change of free energy as a function of guanidine hydrochloride concentration, and **C_m** is the guanidine hydrochloride concentration at which the protein is half-folded.

Now we construct the fitting function fit for use in nonlinear regression.

$$R := 8.314$$

$$T := 300$$

R is the gas constant and T is the temperature of the experiment.

$$\text{fit}(C_m, \text{rate}, C) := \left[\frac{e^{-\frac{\text{rate} \cdot (C_m - C)}{R \cdot T}}}{1 + e^{-\frac{\text{rate} \cdot (C_m - C)}{R \cdot T}}} \right]$$

This function cannot be linearized by a simple transformation, and a nonlinear fit must be performed directly with this function and the data. Remember we are fitting the function to the data. This means that there are adjustable parameters in the function. Our job is to find the best values for these parameters. First we must decide if the function has reasonable values at the limits of the data.

Exercise 2: Show that fit, defined above, has reasonable limits as C approaches zero and as it approaches infinity.

II. INITIAL ESTIMATES OF THE PARAMETERS

In nonlinear least squares regression, initial estimates are required of the parameters. The choice of these initial values are critical, since the fitting routines described below, which are iterative, may not converge or may give physically impossible results (such as a negative concentration) with a poor initial estimate. Some algorithms for fitting functions to data fail if the initial guess is too good. Curve fitting requires patience and some mathematical creativity.

For our case, by observing the graph (see above on page 3), C_m is approximately 2 since this is the value of C in which f would be 0.5. (Note that if $C = C_m$ in the function fit, then $fit = 1/2$. Also, the parameter rate can be estimated by noting that f is near 0.1 when C is 1.

Let $x = e^{\frac{-rate \cdot (C_m - C)}{RT}}$. Then first, use the symbolic processor to solve for x . [My answer is 1/9]. And use the same procedure to solve for C_m .

To obtain the solution write an equation for x and use the Symbolics-Variable-Solve buttons from the Pull-down menu. Before that use the Symbolics-Evaluation Style to set horizontal evaluation steps and showing comments. (Specific details of how this is done vary with the version of Mathcad you are using.) The equation to solve here, based upon the definition of folding fraction, is

$$0.1 = \frac{x}{1 + x} \quad \text{Here 0.1 is the folding fraction and } x \text{ is the equilibrium constant for the unfolded state.}$$

$$e^{\frac{-rate}{R \cdot T}} = (\text{solution_to_x_above}) \quad \text{Solve this for rate, given } x.$$

Thus a good initial estimate of rate is 5500.

Exercise 3: Use the symbolics pull-down menu to perform the above calculations.

III. USE OF A SURFACE PLOT TO DETERMINE INITIAL ESTIMATE OF PARAMETERS

A good method to refine the initial estimates is to use a surface plot. This is performed by determining the standard deviation of fit as a function of the parameters and to plot it near the initial estimates to see where it minimizes.

$$i := 0..6$$

$$SSE(C_m, rate) := \sum_i (f_i - \text{fit}(C_m, rate, C_i))^2 \quad \text{Sum of squares of deviations}$$

Given a set of values for the parameters, SSE indicates how well these parameters fit the data. The goal of nonlinear least squares is to minimize SSE: for a perfect fit, SSE = 0.

Now we produce a 10x10 grid of values of SSE around the initial guess of parameters. The parameters at which SSE is minimum is the set which we wish to determine.

$$k := 0..10 \quad kk := 0..10$$

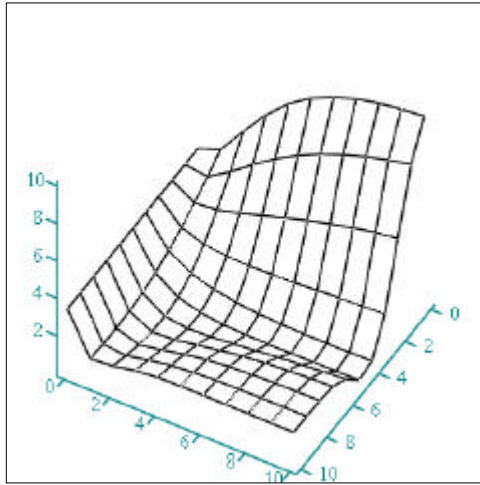
$$XXX_k := 2 + \left(\frac{k-5}{5}\right) \cdot 2 \quad YYY_{kk} := 5500 + \left(\frac{kk-5}{5}\right) \cdot 5500$$

$$ZZZ_{k, kk} := SSE(XXX_k, YYY_{kk}) \quad \text{This produces a 10x10 grid of standard deviations of fit around the initial estimates of the parameters.}$$

$$ZZZ := ZZZ \cdot \frac{10}{\max(ZZZ)}$$

In the following plot, the axis going to the right from the front point (10,10) represents values of C_m and the axis going to the left represents **rate**. Unfortunately, MATHCAD does not use the actual values of the variables but goes from 10 to zero to produce the 10X10 grid.

Thus since **rate** goes from 0 to 11000, then the grid value going to the left must be multiplied by 550 to obtain the value of **rate**. Likewise, the grid value going to the right must be multiplied by 0.4 to obtain the value of C_m .



This graph suggests that C_m is fairly close to 5 (halfway along the axis going to the right from the front point, i.e. the 10,10 point at the front of the figure). The **rate** grid line moving to the left at $C_m = 5$ is not as well-determined due to the flatness of the curve.

ZZZ

Exercise 4 - Rotate the plot by clicking in the box containing the plot and changing the rotation and tilt. Display the plot in color, and see whether the plot is easier or more difficult to interpret.

IV. DETERMINATION OF PARAMETERS

The method used by MATHCAD to find the minimum of a function is called the Levenberg-Marquardt algorithm. This algorithm is explained in nonlinear statistics texts. Basically, given an initial estimate of parameters, the algorithm searches for a direction in parameter space which will minimize the function, using a combination of Newton-Ralfson and steepest descent techniques. Once a minimum along that direction is found, the algorithm finds a new direction and minimizes further, repeating this process until convergence is reached. The convergence criterion may be set using the Mathcad TOL variable.

The algorithm is set up in MATHCAD by using the following steps:

1. Define the function to be minimized (SSE) as a function of the parameters.
2. Give an initial estimate of the parameters.
3. Type the word **Given** (be sure that the word Given is not in text mode.)
4. Type the equation $SSE(\text{parameter1}, \text{parameter2}, \text{etc})=0$
5. Add dummy equations until the number of equations are equal to the number of variables. An example of a dummy equation is $0=0$. Better yet is $m>0$ as a dummy equation.
6. Type **output:=Minerr(parameter1,parameter2,etc.)** (The variable name 'output' may be replaced by any other variable name.)
7. The variable **output** is a vector containing the values of the parameters which minimize the function. The standard deviation is found in **SSE** using the vector **output** for the parameters.

$$\text{fit}(C_m, \text{rate}, C) := \left[\frac{e^{-\frac{\text{rate} \cdot (C_m - C)}{R \cdot T}}}{1 + e^{-\frac{\text{rate} \cdot (C_m - C)}{R \cdot T}}} \right] \quad \text{Function to be fitted}$$

$$\text{SSE}(C_m, \text{rate}) := \sum_i \left(f_i - \text{fit}(C_m, \text{rate}, C_i) \right)^2 \quad \text{Sum of squares of errors that is to be minimized.}$$

Remember that the goal of nonlinear least squares is to find the parameters which minimize SSE. These values for the parameters are the ones that give a close fit between the data and the mathematical function we are using to model the protein behavior.

$$C_m := 2 \quad \text{rate} := 5500$$

Initial estimate of parameters as determined above.

Given

The word Given. Be sure this is not a text word. It must be a Mathcad command.

$$\text{SSE}(C_m, \text{rate}) = 0 \quad \text{rate} > 0$$

Goal of fitting routine; rate > 0 is a constraint equation.

$$\text{output} := \text{MinErr}(C_m, \text{rate})$$

Use of MinErr function to find parameters.

$$\text{output} = \begin{bmatrix} 1.837 \\ 9.894 \cdot 10^3 \end{bmatrix}$$

These are the values of C_m and m from the nonlinear least squares routine.

$$\text{SSE}(\text{output}_0, \text{output}_1) = 0.015 \quad \text{Final SSE}$$

$$\text{SSE}(C_m, \text{rate}) = 0.044 \quad \text{Initial SSE}$$

Exercise 5: Try different initial guesses and see if it affects the value of output. For example, let C_m be negative or zero. Write a summary of your observations in your notebook.

It is always useful to graph the fitted function and see how well it fits the data.

Note the use of different range variables for the raw data and the fitted function.

$$k := 0..6$$

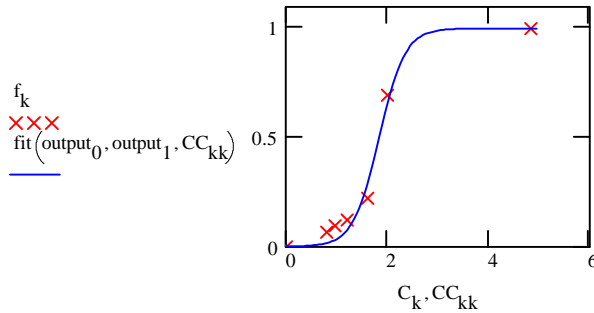
7 data points for raw data

$$kk := 0..99$$

$$CC_{kk} := kk \cdot \frac{5}{100}$$

100 simulated data points for fitted function

Exercise 6 - Modify the last line to use a different number of points for the fitted curve. See how the number of points affects the shape of the fitted curve.



This is the fitted line, along with the original data points.

Exercise 7: Modify the Tolerance variable TOL using the Math-Options pull-down menu and observe the effect on the curve fit.

V. DETERMINING THE UNCERTAINTIES IN THE PARAMETERS

In nonlinear curve fitting the Hessian matrix is used to obtain the standard deviations of the fitting parameters. The elements of the Hessian matrix are defined as

$$H_{i,j} = \frac{1}{2} \left[\frac{d}{d\text{parm}_i} \frac{d}{d\text{parm}_j} (\text{SSE}(\text{parm}_0, \text{parm}_1, \text{etc})) \right] \text{ where SSE has been defined above. For two parameters, } \mathbf{H} \text{ is a } 2 \times 2 \text{ matrix.}$$

The following routine will calculate the Hessian matrix for two parameters. It can be generalized to M parameters.

aa := output₀ bb := output₁

Here we define the elements of the Hessian matrix. Notice the use of the derivatives.

aa := 1.837 *Note: The algorithm for taking derivatives can fail in Mathcad8 if the fitting parameters are 'too precise'. Here we used four significant figures for the value for 'a' to permit evaluation of the derivatives in the Hessian matrix. Try using the output0 from above and see what happens in Mathcad8*

$$H_{0,0} := \frac{1}{2} \frac{d}{daa} \frac{d}{daa} \text{SSE}(aa, bb)$$

$$H_{1,0} := \frac{1}{2} \frac{d}{dbb} \frac{d}{daa} \text{SSE}(aa, bb)$$

$$H_{0,1} := \frac{1}{2} \frac{d}{daa} \frac{d}{dbb} \text{SSE}(aa, bb)$$

$$H_{1,1} := \frac{1}{2} \frac{d}{dbb} \frac{d}{dbb} \text{SSE}(aa, bb)$$

$$H = \begin{bmatrix} H_{0,0} & H_{0,1} \\ H_{1,0} & H_{1,1} \end{bmatrix} \quad H = \begin{bmatrix} 1.612 & 3.801 \cdot 10^{-6} \\ 3.801 \cdot 10^{-6} & 5.746 \cdot 10^{-10} \end{bmatrix}$$

Exercise 8: The Hessian can also be determined by taking SSE and calculating the derivatives symbolically. Perform this action for the above Hessian using the symbolic processor and compare with the above results. (This is a difficult problem.)

The uncertainties are found by using the Hessian matrix with formulas similar to those of linear least squares. A good discussion of the use of the Hessian matrix and its use is given in the text:

Numerical Recipes: The Art of Scientific Computing, by Press, Flannery, Teukolsky, and Vetterling, Cambridge University Press, 1989, P. 529.

Note that the covariance matrix used in **Numerical Recipes** is the inverse of the Hessian matrix.

$$MSE := \frac{SSE(\text{output}_0, \text{output}_1)}{N - 2} \quad MSE = 2.914 \cdot 10^{-3}$$

This is the mean square error, sometimes called "standard deviation of fit".

$$s_{\text{output}_0} := \sqrt{MSE \cdot (H^{-1})_{0,0}} \quad s_{\text{output}_1} := \sqrt{MSE \cdot (H^{-1})_{1,1}}$$

These are the uncertainties of the parameters.

Exercise 9 - Compare these equations with those used in linear least squares. What is different about these?

The results with their uncertainties are as follows:

$$\text{output} = \begin{bmatrix} 1.837 \\ 9.894 \cdot 10^3 \end{bmatrix} \quad s_{\text{output}} = \begin{bmatrix} 0.043 \\ 2.27 \cdot 10^3 \end{bmatrix}$$

Recall that 'output' was evaluated two pages above in this document.

To test for the significance of the parameters, divide the parameter by the standard deviation of parameter, and compare with Student's t at the 95% significance level. If the ratio is greater than the value of t, then the parameter is significant.

Mathcad can calculate the student t-test values. To do this use $qt(\{1 - (1-\text{range})/2\}, \{N-2\})$ to obtain a value for Student's t, where **range** is the confidence interval (for example, 95%), and **N** is the number of data points. Alternatively, the value of Student's T may be found in most statistics books. Calculation of the t-text values here is, of course, more convenient.

$$\frac{\text{output}_0}{\text{soutput}_0} = 42.855$$

$$qt(0.975, N - 2) = 2.571$$

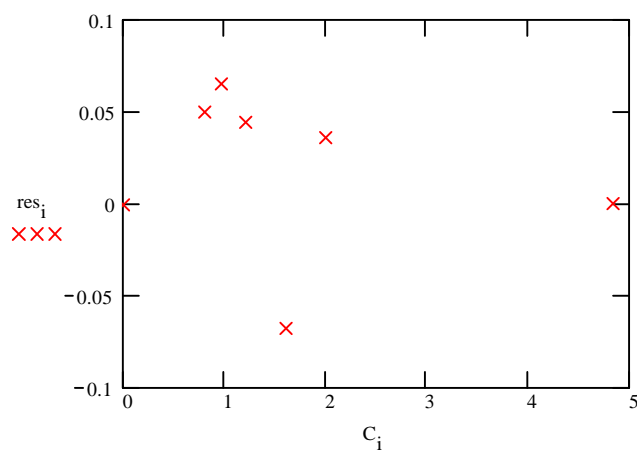
Exercise 10 - Show that output1 is statistically significant for this example. Repeat the calculation at the 99% confidence level. Draw a conclusion about the degree of confidence in the fitting parameters in this experiment.

VI. RESIDUAL ANALYSIS

The residual of a data point is the difference between the experimental and calculated value of y . Here we will calculate residuals and use them to discover potential points to discard from our data set.

$$\text{res}_i := (f_i - \text{fit}(\text{output}_0, \text{output}_1, C_i))$$

Calculating the residuals, **res**, and storing them in a vector.



Graph of **res** vs. **C**

Note that all but one the residuals are positive. This point, the 4th in the **res** vector using standard Mathcad 0 to 6 numbering scheme for this vector.

Point 4 is also far away from zero and is a possible candidate for removal. Thus, we shall repeat the calculations done above in this document with the questionable point removed. First let's use an established statistical technique to test if point 4 should be rejected.

Exercise 11 - Use Chauvenet's criterion to determine whether point 4 should be statistically removed from the fit. See the companion document "Linear Least Squares Regression" for an example of the use of Chauvenet's criterion. (Young, S. H. ; Wierzbicki, A. J. *Chem. Educ.* 2000, 77, 669.)

All of the calculations that follow are a repeat of what was done earlier in this document. Therefore the annotation level is much lower. Students are encouraged to annotate this material as a way of improving understanding and retention of concepts contained here.

Use the suffix REV for the revised variables.

$iii := 0..3$ $CREV_{iii} := C_{iii}$ $fREV_{iii} := f_{iii}$ Redefine matrix **f** for N-1 data points, removing point
 $iii := 5..N-1$ $CREV_{iii-1} := C_{iii}$ $fREV_{iii-1} := f_{iii}$ 4. Review: **f** was the original set of folding
 $iREV := 0..N-2$ fraction parameters.

$$SSEREV(Cm, m) := \sum_{iREV} \left(fREV_{iREV} - \text{fit}(Cm, m, CREV_{iREV}) \right)^2$$

$Cm := 2$ $m := 5500$

Given

$SSEREV(Cm, m) = 0$ $m > 0$

$outputREV := \text{MinErr}(Cm, m)$

$$outputREV = \begin{bmatrix} 1.754 \\ 8.036 \cdot 10^3 \end{bmatrix}$$

Compare outputREV to output. What is the change if any? What is the significance of the change? Redraw the fitting function using the outputREV parameters and compare to the experimental data. Is the fit improved? Record your observations in your notebook.

Next recalculate the Hessian matrix.

$$\text{aaREV} := \text{outputREV}_0 \quad \text{bbREV} := \text{outputREV}_1$$

$$\text{HREV}_{0,0} := \frac{1}{2} \frac{d}{d \text{aaREV}} \frac{d}{d \text{aaREV}} \text{SSEREV}(\text{aaREV}, \text{bbREV})$$

$$\text{aaREV} := 1.754$$

$$\text{HREV}_{0,1} := \frac{1}{2} \frac{d}{d \text{aaREV}} \frac{d}{d \text{bbREV}} \text{SSEREV}(\text{aaREV}, \text{bbREV})$$

Note the need to retype the aaREV in Mathcad8

$$\text{HREV}_{1,0} := \frac{1}{2} \frac{d}{d \text{bbREV}} \frac{d}{d \text{aaREV}} \text{SSEREV}(\text{aaREV}, \text{bbREV})$$

$$\text{HREV}_{1,1} := \frac{1}{2} \frac{d}{d \text{bbREV}} \frac{d}{d \text{bbREV}} \text{SSEREV}(\text{aaREV}, \text{bbREV})$$

$$\text{HREV} = \begin{bmatrix} 0.716 & 2.755 \cdot 10^{-6} \\ 2.755 \cdot 10^{-6} & 1.809 \cdot 10^{-9} \end{bmatrix}$$

Note that removal of one point changes the Hessian matrix. The original Hessian matrix was

$$\text{H} = \begin{bmatrix} 1.612 & 3.801 \cdot 10^{-6} \\ 3.801 \cdot 10^{-6} & 5.746 \cdot 10^{-10} \end{bmatrix}$$

$$\text{SSEREV}(\text{outputREV}_0, \text{outputREV}_1) = 1.699 \cdot 10^{-3} \quad \text{SSE}(\text{output}_0, \text{output}_1) = 0.015$$

How has the removal of point 4 changed the sum of squares value? Is the situation for the fitting improved? Does the graph you prepared with the data and the fitted function look different? Explain.

$$\text{MSEREV} := \frac{\text{SSEREV}(\text{outputREV}_0, \text{outputREV}_1)}{N - 3} \quad \text{Divide by N-3 here: N-1 data points and 2 parameters}$$

$$\text{MSEREV} = 4.246 \cdot 10^{-4} \quad \text{MSE} = 2.914 \cdot 10^{-3}$$

What effect do you see on the MSE by removal of 1 data point?

$$\text{soutputREV}_0 := \sqrt{\text{MSEREV} \cdot (\text{HREV}^{-1})_{0,0}} \quad \text{soutputREV}_1 := \sqrt{\text{MSEREV} \cdot (\text{HREV}^{-1})_{1,1}}$$

$$\text{soutputREV} = \begin{bmatrix} 0.024 \\ 485.964 \end{bmatrix} \quad \text{soutput} = \begin{bmatrix} 0.043 \\ 2.27 \cdot 10^3 \end{bmatrix}$$

Again, what effect is there on the uncertainties of the parameters by removal of just 1 data point?

Conclusion: Removal of the questionable data point causes the MSE to dramatically decrease and the uncertainties in the parameters to decrease significantly. This indicates that the questionable point should be removed from the data set.

Exercise 12: Is it possible that removing a data point could cause the value of MSE to *increase*?

VII. SUMMARY

This template has demonstrated many statistical concepts, all of which are useful when fitting to a nonlinear curve. The following problem should be performed on a new Mathcad page, demonstrating many of the principles described above.

Exercise 13: Use the data set nonlinear.prn, which is a data set consisting of x,y pairs of fluorescence intensity as a function of concentration of guanidine hydrochloride for the unfolding of a protein. Fit this data to the Michaelis-Menton

function $Y_i = \frac{\text{THETA1} \cdot X_i}{\text{THETA2} + X_i}$ where THETA1 and THETA2 are fitting parameters, to be

determined. The following steps provide a guide for you to complete the fitting procedure:

- a) **determine initial estimates for THETA1 and THETA2**
- b) **compute the function SSE (standard deviation of fit as a function of X, Y)**
- c) **obtain the values of the fitting parameters using nonlinear regression (either the solve-block Minerr method or the genfit function)**
- d) **compute the standard deviation of the fit**
- e) **create the Hessian matrix**
- f) **compute the standard deviation of the parameters**
- g) **Use the Student's t test to determine validity of the parameters**
- h) **Draw the Residual plot**
- i) **Finally, if any bad points exist, observe the effect of their removal after statistically determining if they should be removed.**