X. NONLINEAR REGRESSION

WHAT IS THE OBJECTIVE

There are essentially two different objectives:

1. To fit data where the equation structure and equation parameters are not of interest. The objective is to have a smooth analytical function that fits the data for interpolation and possible projections.
2. To obtain parameter values for a given equation, where the accuracy of the parameter values are important.

There are only three methods in Excel:

1. Transforming the equation and variables so that a linear fit (i.e. LINEST) is possible on the transformed variables and then working backwards to construct the non-linear equation from the LINEST output.
2. Fit by use of Solver.
3. Fit by use of an Excel add-in.

NON-LINEAR EQUATION FITTING:

THE GENERAL PROCESS

For non-linear equation solving, a minimizing routine is the primary method of fitting. The method is numerical, and there are many methods to compute a systematically-improving-estimate of equation parameters. A measure of the lack of fit is computed and iterations are repeated until some fit criteria is attained. One of the major problems here is about global or local fits, where the residual surface may have many local minimums but only one global minimum. Non-linear equation fitting will easily end in local minimums.

Non-linear equation fitting is a trial-and-error process. One tries different starting parameter values, evaluates the outcome, resets internal operating parameters (limits, tolerance levels, manual or automatic methods, etc.) At some point there is a decision to stop the process and accept the best results. Chi-square (and F) tests on the sum-of-squares can help in obtaining p values for making a decision on accepting a solution.

One method to efficiently do this process is to analytically calculate derivatives (gradient vector, g(x)) and to obtain estimates of the second derivatives (Hessian matrix, G(x)) if the routine is to have a high degree of success on a wide variety of problems. The approach may be to (a) set up equations sets that directly calculate g(x) and G(x) at a given set of parameter values or (b) internally approximate the non-linear response surface by an approximating quadratic, and calculate g(x) and G(x) approximations. Method b fails when the higher order terms (which are assumed to be negligible) cannot be neglected.

In general there are two classes of fitting problems, one is the general unconstrained fit, in which there are no constraints on the fitted parameters, and the other is the constrained fit, where constraints are put on fitted parameter values, such as being positive, being greater than a specified limit, etc. The unconstrained methods are the ones evaluated in this section. The test data, the models and the criteria are all in the unconstrained class.

THE BASIC NIST NON-LINEAR EQUATION FIT TEST SUITE

The NIST test is on the fitting-software being able to fit sets of data to a specified equation. These are all unconstrained fit problems. Table 11-1 shows the basic data set.

Table 11-1: NIST StRD Data Set Information

Sequence	Dataset	Class	Difficulty	Size	Number of Significant Figures	Number of X Variables	Number of Coefficients	Number of Exponential Terms[1]
38	Misrala	Simple Exponential	1	14	4	1	2	1-1p
39	Chwirut2	Simple Exponential	1	54	4	1	3	1-1p
40	Chwirut1	Simple Exponential	1	214	5	1	3	1-1p
41	Lanczos3	Complex Exponential	1	24	5	1	6	3-1p
42	Gauss1	Complex Exponential	1	250	7	1	8	1-1p, 2-2p
43	Gauss2	Complex Exponential	1	250	7	1	8	1-1p, 2-2p
44	DanWood	Algebraic	1	6	4	1	2	
45	Misra1b	Algebraic	1	14	4	1	2	
46	Kirby2	Algebraic	2	151	5	1	5	
47	Hahn1	Algebraic	2	236	5	1	7	
48	Nelson	Simple Exponential	2	128	3	2	3	1-1p
49	MGH17	Simple Exponential	2	33	3	1	5	2-1p
50	Lanczos1	Complex Exponential	2	24	13	1	6	3-1p
51	Lanczos2	Complex Exponential	2	24	6	1	6	3-1p
52	Gauss3	Complex Exponential	2	250	7	1	8	1-1p, 2-2p
53	Misra1c	Algebraic	2	14	4	1	2	
54	Misra1d	Algebraic	2	14	4	1	2	
55	Roszman1	Trig Function	2	25	6	1	4	
56	ENSO	Trig Functions	2	168	3	1	9	
57	MGH09	Algebraic	3	11	3	1	4	
58	Thurber	Algebraic	3	37	7	1	7	
59	BoxBOD	Simple Exponential	3	6	3	1	2	1-1p
60	Rat42	Simple Exponential	3	9	4	1	3	1-2p
61	MGH10	Simple Exponential	3	16	4	1	3	1-2p
62	Eckerle4	Simple Exponential	3	35	7	1	3	1-2p
63	Rat43	Simple Exponential	3	15	5	1	4	1-2p
64	Bennett5	Algebraic	3	154	8	1	3	

The equations to be fitted to the NIST StRD data sets are given with each data set. They are based on some theory on which the data and how it was obtained (e.g. from a very specialized scientific instrument). They essentially are of scientific and engineering

[1] The first number is the number of separate exponential terms. The second number is the number of parameters to be fitted in each term.

interest only. The equations are given in Table 11-2 to show the complexity of the fitting requirement.

Table 11-2: NIST Equations to be Fitted

Sequence	Equation[2]
	y = b1*(1-exp[-b2*x]) + e
39	y = exp(-b1*x)/(b2+b3*x) + e
40	y = exp[-b1*x]/(b2+b3*x) + e
41	y = b1*exp(-b2*x) + b3*exp(-b4*x) + b5*exp(-b6*x) + e
42	y = b1*exp(-b2*x) + b3*exp(-(x-b4)^2 / b5^2) + b6*exp(-(x-b7)^2 / b8^2) + e
43	y = b1*exp(-b2*x) + b3*exp(-(x-b4)^2 / b5^2) + b6*exp(-(x-b7)^2 / b8^2) + e
44	y = b1*x^b2 + e
45	y = b1 * (1-(1+b2*x/2)^(-2)) + e
46	y = (b1 + b2*x + b3*x^2) / (1 + b4*x + b5*x^2) + e
47	y = (b1+b2*x+b3*x^2+b4*x^3) / (1+b5*x+b6*x^2+b7*x^3) + e
48	log[y] = b1 - b2*x1 * exp[-b3*x2] + e
49	y = b1 + b2*exp[-x*b4] + b3*exp[-x*b5] + e
50	y = b1*exp(-b2*x) + b3*exp(-b4*x) + b5*exp(-b6*x) + e
51	y = b1*exp(-b2*x) + b3*exp(-b4*x) + b5*exp(-b6*x) + e
52	y = b1*exp(-b2*x) + b3*exp(-(x-b4)^2 / b5^2) + b6*exp(-(x-b7)^2 / b8^2) + e
53	y = b1 * (1-(1+2*b2*x)^(-.5)) + e
54	y = b1*b2*x*((1+b2*x)^(-1)) + e
55	y = b1 – b2*x - arctan[b3/(x-b4)]/pi + e

[2] The equation as set up within Excel cells is different from that shown in table 11-2. The Excel form has added parentheses to account for Excel's calculation precedences. These are different from the mathematical precedences, and if not done, will come out with wrong numbers, See Berger 2007.

Sequence	Equation
56	y = b1 + b2*cos(2*pi*x/12) + b3*sin(2*pi*x/12) + b5*cos(2*pi*x/b4) + b6*sin(2*pi*x/b4) + b8*cos(2*pi*x/b7) + b9*sin(2*pi*x/b7) + e
57	y = b1*(x^2+x*b2) / (x^2+x*b3+b4) + e
58	y = (b1+b2*x+b3*x^2+b4*x^3) / (1+b5*x+b6*x^2+b7*x^3) + e
59	y = b1*(1-exp[-b2*x]) + e
60	y = b1 / (1+exp[b2-b3*x]) + e
61	y = b1 * exp[b2/(x+b3)] + e
62	y = (b1/b2) * exp[-0.5*((x-b3)/b2)^2] + e
63	y = b1 / ((1+exp[b2-b3*x])^(1/b4)) + e
64	y = b1 * (b2+x)^(-1/b3) + e

The NIST StRD sets give you a hint on how to start, by giving two sets of starting values. These are close enough to the global minimum and have slopes at these starting points in the direction of the minimum. McCullough (2000) concludes that “Start I” which is far from the solution should be the starting point for any testing. However “Start II” would reduce the possibility of being trapped at a local minimum. In real life problems, good starting points are not known, and there is a strong likelihood of being trapped at a local minimum, or being in a very shallow well, where the final solution varies depending on the starting point.

McCullough (2000) points out that the NIST solutions are not necessarily unique, global-sum-of-error-squares-minimums, and that “other” solutions having “low” sum-of-error-square values exist. However the literature does not identify these other “points” for the NIST non-linear test suite. These other “points” are assumed to be errors in the software solution method.

The NIST source only gives 11 decimal digits for the correct value of the equation coefficients, and therefore an LRE value of 11 or higher is considered a perfect fit.

NON-LINEAR EQUATION FITTING IN EXCEL WITH SOLVER

THE GENERAL SOLVER PROCESS

Solver uses numerically calculated differences about a given point as estimates of g(x). (Microsoft KBA 82890).

In general, using Solver to fit non-linear regressions, is to set up a worksheet with the X and Y values in columns A and B. Then use intermediate columns to obtain a calculated Y, given values for the equation. From the Y column, generate residuals and residuals squared columns. At the next column put in the starting values for the constants of the equation, and below the set, put in the formula “=SUM(.. range of the residuals squared column…)”. This is the cell that solver will minimize. This cell should be in focus when Solver is called. The first menu box sets up the application and constraints, and the second, the options menu sets up the limits and search ending criteria. The Help screen describes what all the inputs are.

Choice of the starting values of the parameters is important, since Solver only goes to a local minimum nearest to the starting values. Solver is not a global optimizer. There are internal routines that determine when Solver stops the iterations and concludes that a solution has been obtained.

Solver doesn't provide any associated uncertainty estimates. Solver-Aid displays standard deviations, the covariance matrix, and (optionally) the matrix of linear correlation coefficients.”

For equation fitting, the only method here is to evaluate at a stopping point, the residuals and the sum-of-squares of the residuals. McCullough (2003) says that “The testing requires access to the gradient (which Solver computes but does not allow the user to access, the Hessian (which Solver does not compute), and the trace (which Solver computes but does not store so the user has to copy it by hand, a laborious task)”.

REPORTED PROBLEMS ON NON-LINEAR REGRESSION

There was not much reported on the use of Solver in the statistical literature. Miles (2005) reported on its use to solve simple structural equation problems. The bulk of the literature was about the failure of Solver to fit the NIST non-linear equations.

De Levie (2005) has found Solver to work sometimes. “Solver often yields a result that, upon rerunning it a second time, is slightly improved. If the answer matters, it may therefore be wise to run Solver twice in a row. Solver often yields good results, but can be quite far off in pathological cases, such as described by B. D. McCullough & B. Wilson in Computational Statistics and Data Analysis 31 (1999) 27.”

Table 11-3: Excel Faults in Non-linear Regression

Application or Function	Problem	Source	Fix or Comments
Regression, Non-linear using Solver	Accuracy and unable to arrive at a solution	McCullough, Cox 2000, Mondragon and Borchers (2005):	Excel works for the simpler models. May require model simplification, re-parameterization, sequential regression or other starting values.
Regression, Non-linear using Solver	Accuracy of solution cannot be determined. No standard errors provided.	McCullough, Cox 2000	See Comments below.

TEST RESULTS:

The Solver application evaluated here is fitting non-linear equations to StRD data sets (Table 11-2).

The test results reported below are from McCullough and Wilson (1999 and 2000). The LRE values are from their table 8, which represents the best that they could get Solver to do. Zero LRE values represent situations that include Solver stopping away from a local minima. Refer to the original papers for more information on the testing methods. The JMP values come from Creighton and Ding (2002).

Table 11-4 gives the reported accuracy of the values of the calculated coefficients using the SOLVER routine. M&W is from McCullough and Wilson (1999). M&B is from Mondragon and Borchers (2005). In both cases it is the lowest LRE value for the complete set of equation coefficient values.

Table 11-4: Test Results Reported, LRE Value for Coefficients

Sequence	M&W, Excel Lowest LRE	M&B, Excel Lowest LRE	M&B, Excel Highest LRE	M&W Corrected Assessment
38	4.8	4.8	6.1	Pass
39	4.6	4.2	4.6	Pass
40	4.9	4.0	4.9	Pass
41	0.0	0.0	0.0	Fail
42	0.0*[3]	4.6	4.7	Pass
43	0.0*	4.4	4.5	Pass
44	5.5	4.6	4.7	Pass
45	4.4	4.4	6.4	Pass
46	1.1	1.0	1.9	Pass
47	0.0	0.0	0.0	Fail
48	1.3	0.0	0.0	Pass
49	0.0	0.0	1.4	Fail
50	0.0	0.0	0.0	Fail
51	0.0	0.0	0.0	Fail
52	0.0*	4.1	4.3	Pass
53	4.6	0.0	0.0	Pass
54	5.3	4.4	5.2	Pass
55	3.7	0.0	3.5	Pass
56	3.4	0.0	0.0	Pass
57	0.0	0.0	5.0	Fail
58	1.8	1.5	1.7	Pass
59	0.0	0.0	5.6	Fail
60	[illegible]	5.2	5.3	Pass

[3] The three Gauss series(*) results were reevaluated by McCullough (2008) based on Berger (2008). If the equations are rewritten to fit the Excel calculation precedences, the three Gauss LRE values usimg Solver are now 5.2, 4.9 and 4.3. These values are in agreement with Mondragon and Borchers (2005).

Sequence	M&W, Excel Lowest LRE	M&B, Excel Lowest LRE	M&B, Excel Highest LRE	M&W Corrected Assessment
61	0.0	0.0	0.0	Fail
62	0.0	0.0	5.1	Fail
63	0.0	0.0	3.2	Fail
64	0.0	0.0	0.0	Fail

The M&W values in column 1 come from starting point I, using automatic scaling and a convergence criteria of 1E-07. If start I failed, then start II was used. The specific starting point for each M&W reported value was not identified.

The M&B lowest and highest values come from the two NIST starting value options. M&B report that the lowest and highest values that they found did not consistently relate to starting point I or II. Berger (2006 and 2007) points out that the specific way the cell equation is constructed has an effect on the solution. This has to do with the internal Excel precedence's on doing additions, subtractions, multiplications, divisions, and exponentiations; and in the manner in which the + and – signs are located. This is discussed below.

Out of 27 cases, Solver was only consistently able to get close to a solution (i.e. LRE values >4.0) in 11 cases. For 4 cases (48, 53, 55 and 56) there was no consistent solution (i.e. M&W results differed from M&B results). There were 11 cases that had no acceptable Solver solution. and were give a "Fail" designation. This gives Solver a grade of 59%, which is not considered passing.

THE EXCEL PRECEDENCE PROBLEM

One of the indefined critical problems in regard to equation accuracy has to do with how the equation is formed in Excel. This is the precedence problem (see………..)

Berger (2006 and 2007) points out that negation has a higher order of precedence than exponentiation, multiplication, division, addition and subtraction. The way that the equation to be fitted is set in by using parentheses, can change the way this precedence operates and gives computed values.

"To see this in Excel, enter = -3^2 into a cell. The result Excel reports is +9 because Excel interprets this as (-3)^2. On the other hand, = 0 - 3^2 results in -9, because now the "-" sign is interpreted as subtraction (lower order of precedence that exponentiation) rather than negation."

"Consider the models Gauss1, Gauss2, and Gauss3 in the StRD datasets. They have terms like exp(-(x-b4)^2/b5^2). If this term is entered in this way when calculating predicted values, Solver will obtain the wrong answer, because Excel interprets this as exp((-(x-b4))^2/b5^2) (note extra parentheses). If this and similar terms are entered as exp(-((x-b4)^2)/b5^2) then Solver achieves the correct solution to 6 or more decimal places even with the default settings of convergence criteria. I suspect that when McCullough and Wilson conducted the tests, this term was entered as exp(-(x-b4)^2/b5^2), resulting in zero decimal place accuracy.".

Berger states that, "This feature was discovered by two high school students, Donald Brandl and Jebina Rajbhandari, who worked on a summer project with me in 2000. I have never seen it mentioned in your papers or anywhere else. Anyone who uses formulas in Excel needs to know about it."[4]

USING SOLVER TO FIT EQUATIONS TO DATA

In any case where Solver is used, one should try different options, such as tangent and quadratic estimates, forward and central derivatives, and Newton and conjugate search methods. Sometimes Solver seems to arrive at a better solution (smallest sum-of-squares value) directly from a starting set with a small convergence criterion, and sometimes a stepped approach, with difference coarse convergence setting and a final fine convergence setting. However you cannot depend on the final parameter values as being an accurate solution.

One cannot be sure that the resulting equation parameter values have any accuracy. When Solver works, we can generally be sure of 2 significant decimal digits, but we have no assurance that Solver worked correctly, or correctly found the global maximum/minimum.

The number of parameters to be fitted should generally not exceed 3. The number of data points should be greater than 20. The equations should not have strong interactions between coefficients such as products, powers or quotients of coefficient terms. These restrictions reduce the occurrences of local "near" minima, and Solver getting hung up on false minima. Excel does not retain a record of your progress, so you have to manually put necessary information on the worksheet. Set up separate cells with all the information relating to each trial, starting values, settings, outputs, number of iterations, reason for the stop, sum-of-squares of the residuals, F test p values, etc. Any residual plots should be on the worksheet below the work area.

Because of the use of numerical differencing for derivatives, numerical accuracy and the tolerance settings (used when there are parameter constraints) have to be set low or numerical instability occurs. This has been noted on the Excel discussion lists.

There are others who have had better success using Solver to find coefficient values for the NIST non-linear equation suite. These other sources, appear to be more experienced in using Solver and their results are somewhat better than that reported by McCullough and Wilson and Mondragen and Borcher. Tables 11-5A and 11-5B are a summary of some of the other approaches to solving the NIST MGH17 problem (number 49) that failed.

---

[4] This characteristic appears to be well known, see <www.macnauchtan.com/prb/precedence.html>

Table 11-5A A Summary of Other References on the Use of Solver, The NIST MGH17 Data Set and Equation.

ROW	Source	Tool
1	Billo (2007)	Solver
2	Billo (2007)	Solver
3	Billo (2007)	Solver
4	Billo (2007)	Solver
5	Billo (2007)	Solver
6	Billo (2007)	Solver
7	Billo (2007)	Solver
8	Billo (2007)	Solver
9	Billo (2007)	Solver
10	Billo (2007)	Solver
11	Creighton & Ding	JMP Non-Linear Platform
12	Heiser	Levenberg-Marqurdt
13	Heiser	Nelder-Mead Downhill Simplex
14	McCullough & Wilson (1999), Table 8	Solver
15	McCullough & Wilson (1999), Table 8	Solver
16	McCullough & Wilson (1999), Table 8	Solver
17	McCullough & Wilson (1999), Table 8	Solver
18	Mondragen & Borcher	Solver
19	Mondragen & Borcher	Solver

Table 11-5B. A Continuation of Table 5-11A on the MGH17 Problem

ROW	Trial	Method	b1	b2	b3	b4	b5	low	High
1	1	Default Solver parameters	2.11	0.82	0.67	1.31	1.26		
2	2	Use Automatic Scaling	2.00	0.72	0.57	1.21	1.16		
3	3	Objective / 10^-5	2.70	1.31	1.19	1.84	1.76		
4	4	Same as trial #3, plus Automatic Scaling	2.00	0.72	0.57	1.21	1.16		
5	5	Same as trial #3, plus convergence =0.000001	2.70	1.31	1.19	1.84	1.76		
6	6	Same as trial #3, using central derivatives	2.79	1.40	1.29	1.93	1.86		
7	7	Same as trial #3, using quadratic estimates	2.70	1.31	1.19	2.70	1.76		
8	8	Same as trial #6, using quadratic estimates, plus convergence =0.000001	2.79	1.40	1.29	1.93	1.86		
9	9	Same as trial #6, but re-solving	5.95	4.67	4.55	5.16	5.15		
10	10	Same as trial #3, but re-solving	2.70	1.31	1.19	1.84	1.76		
11		Gauss-Newton, analytic derivatives, stepped tolerances	11.0	11.0	11.0	11.0	10.8		
12	1	Analytic derivative function input, start 1	11.09	11.20	11.04	11.35	10.83		
13	1	6 repeats to stable value set, start 1	8.87	7.61	7.49	8.10	8.06		
14	A	Default Estimation						0	
15	B	Convergence Tolerance set to 1E-07						0	

16	C	Automatic Scaling						0	
17	D	Convergence Tolerance set to 1E-07 and Automatic Scaling						0	
18	1	Quasi-Newton, automatic scaling, tolerance of 1E-15, Start 1						0	0
19	2	Quasi-Newton, automatic scaling, tolerance of 1E-15, Start 2						0	1.4

Billo’s (2007) values suggest that experience with Solver and with the different options and settings can result in an improvement in the accuracy of the solution.

Billo (2007) says, “In my experience, it is most important to manually scale the objective (sum of squares) so as to obtain a value near unity. This is independent of ‘Use Automatic Scaling’, which as far as I can determine, scales the changing-cells. In addition, although I rarely switch from forward to central derivatives, I found that changing that decreased the sum-of-squares in some cases, as did decreasing convergence, or "re-solving" after the Solver had reached its original convergence.” (See also Billo 2001)

Billo (2007) said, ‘I was able to get fairly satisfactory fits to the data sets that I chose (table below).

Table 11-6: Comparisons, Billo’s LRE Results With Table 11-4

Sequence	Data Set	Lowest LRE	Highest LRE	M&W Low	M&B Low	M&B High
42	Gauss1	8.6	9.3	-	4.6	4.7
49	MGH17	4.5	5.9	0	0	1.4
57	MGH09	9.7	10.2	0	0	5.0
61	MGH10	2.5	3.6	0	0	0
64	Bennett5	2.7	3.4	0	0	0

Values from table 11-4 were included in table 11-6 to provide a comparison of results. This shows that it is possible to get better results from Solver by using his recommendations.

McCullough reaffirms “The only reason Billo (or anyone else) knows they've found a solution to any non-linear least squares fitting problem is if they already know the answer. Solver stops at so many non-solutions and has no diagnostic capabilities, so there really is no way to differentiate the non-solutions from the solution EXCEPT by knowing the answer in the first place.”

The essential-diagnostic however still remains, that of using the Solver returned coefficient values to calculated predicted values, and then by calculating the RMSE difference from the data. Under this modus, one can try different Solver inputs such as described by Billo and others, and view the resulting RMSE values. However we still have that uncertainty when the minimum RMSE value is obtained, about equation parameter values. Acceptance of fitted parameter values then does not come from LRE values but from the lowest RMSE value. This does not however allow one to estimate how far the resulting parameter values are from a fit using better and more reliable methods, (such as the Levenberg-Marquardt method). The lowest RMSE value may still be a zero LRE value.

The reported results in tables 11-4, 11-5B and 11-6 on the use of Solver, show considerable variation in fit reported by the different authors. This clearly indicates a real problem on the reliability and dependability of Solver results. This is why Solver should be avoided, since there is no way to know that a reliable solution was obtained from the RMSE value. The minimum RMSE value is not an indicator that the calculated coefficient values are a reliable solution, but they may be an “acceptable” solution.

A look at table 11-4, suggests the types of non-linear equations that Solver can handle. These are the simpler 2-3 parameter equations involving exponentials, simple ratios, simple sums and little interactions between coefficients. They are also problems where m is much greater than n. These are equations 38, 39, 40, 44, 45, 54 and 60. Re-trying a Solver solution for cases 42, 43, 52, 53, 55 and 56 by trying different starting values does result in LRE values above 3. However without a known objective value (such as RMSE values), one does not know which trial improved the fit.

The cell equation should also be appropriately set up to accommodate Excel computing precedence’s. This is discussed below.

Table 11-7 shows results from other software programs as a comparison to the Excel results. Column 4, the JMP data is from Creighton and Ding (2000) which reported LRE values for each of the fitted coefficients, in the order b1, b2, b3…, where the b’s are identified in Table 11-3.

Table 11-7: Test Results Reported, LRE Value for Coefficients

Sequence	M&B, MATLAB Highest LRE	M&W, Stata Lowest LRE	JMP 4.0.5 Nonlinear Platform
38	11.0	9.1	11.0, 11.0
39	10.6	7.9	10.9, 11.0, 10.9,
40	10.3	7.6	9.9, 10.6, 10.3
41	5.1	6.2	11.0, 11.0, 11.0, 11.0, 10.5, 11.0
42	6.9	8.6	11.0, 10.9, 11.0, 11.0, 10.8, 11.0, 10.8, 10.6,
43	6.8	8.2	11.0, 10.5, 10.3, 10.4, 10.7, 11.0, 11.0, 10.8
44	10.2	8.6	11.0, 11.0
45	11.0	8.3	11.0, 11.0
46	10.4	9.1	9.9, 9.9, 10.3, 10.0, 10.9
47	9.7	7.1	11.0, 11.0, 10.9, 11.0, 11.0, 11.0, 10.8
48	0.0	7.1	10.9, 10.1, 11.0
49	0.0	9.4	11.0, 11.0, 11.0, 11.0, 10.8
50	5.8	10.6	11.0, 10.6, 11.0, 10.9, 10.6, 11.0
51	5.7	7.4	11.0, 10.4, 11.0, 11.0, 10.8, 11.0
52	6.5	8.2	11.0, 11.0, 10.7, 10.6, 11.0, 11.0, 10.5, 10.7
53	10.8	9.2	11.0, 10.8
54	11.0	[illegible]	[illegible]

Sequence	M&B, MATLAB Highest LRE	M&W, Stata Lowest LRE	JMP 4.0.5 Nonlinear Platform
55	4.0	7.9	11.0, 11.0, 10.9, 11.0
56	6.6	4.7	10.7, 10.9, 11.0, 11.0, 10.8, 11.0, 11.0, 11.0, 10.7
57	5.2	7.0	8.9, 7.5, 8.1, 7.7
58	7.8	6.5	10.4, 10.5, 11.0, 11.0, 11.0, 11.0, 11.0
59	9.7	7.3	10.5, 9.8
60	11.2	7.6	11.0, 11.0, 11.0
61	0.0	7.5	11.0, 11.0, 10.9
62	8.1	8.3	10.6, 10.7, 11.0
63	1.3	6.0	11.0, 11.0, 11.0, 11.0
64	3.7	6.3	11.0, 11.0, 11.0

Table 11-7 shows that JMP 4.0.5 can do an exceptional job in fitting non-linear equations. The success is from the algorithm that calculates both first and second analytical derivatives of the fitting function. They also used a sequence of tolerance values from 1E-07, 1E-12 and 1E-30. The software also allows the routine to continue running even after the tolerance level was met. They just continued running the program until the parameter values did not change (Creighton and Ding 2002),

There were no changes to the Solver add-in in any of these versions. The results of tests on the 97 and 2000 versions reported above are valid for all versions since then.

NON-LINEAR EQUATION FITTING IN EXCEL WITH EXCEL ADD-INS

This represents a class of available functions and routines to the basic Microsoft Excel that provides extended optimization and non-linear equation fitting capabilities. Only two free packages were evaluated, since the requirement here was that the package be freely downloaded, and the VB code not password protected. There are many free add-ins, but they are all password protected, so the method of solution is unknown, and assumed to be irregular and highly error prone. The inability to establish any dialog between the source and the tester was also a consideration.

THE LEVENBERG - MARQUARDT SET

This is a method of fitting non-linear equations to data with no constraints on the resulting parameter values. It is not an optimization method, but can be used for optimizations. It does give accurate parameter values for all of the NIST StRD equations described in table 11-3 above. It is a free VBA add-in package[5] from the Foxes Team

[5] See Note XN on the Levenberg-Marquardt Algorithm. The URL http://digilander.libero.it/foxes/Documents and select Levenber- Marquardt VB routine. This downloads a 4 page document that describes the method and how to use it. There are two *.bas files that are then to be downloaded, and the tutorial shows how to build the function and derivative equations in VB. Also

(Volpi 2006b) that will return accurate parameter values. It does require knowledge of VBA, since a user has to develop an overall calling subroutine and a set of function in VBA. A function describing the non-linear equation, and another function giving the first derivative of the equation is also required. It requires knowledge of calculus to develop these equations. The accuracy of the results is dependent on the accuracy and robustness of the equation sets. The current version[6] is only able to fit one independent variable (x) value.

This method does very well on the NIST StRD data, as shown in table 11-8

Table 11-8: LRE Values From Garcia's Levenberg - Marquardt Algorithm

NIST Number	NIST Name	Fitted Coefficient LRE Values
38	Misrala	11.38, 11.09
39	Chwirut2	10.93, 11.28, 10.85
40	Chwirut1	10.72, 11.75, 11.25
41	Lanczos3	11.56, 11.39, 11.24, 11.47, 10.50, 12.73
42	Gauss1	12.51, 10.79, 11.36, 11.32, 10.83, 11.43, 10.82, 10.38
43	Gauss2	12.19, 10.48, 10.37, 10.42, 10.74, 11.30, 11.08, 10.76
44	DanWood	11.25, 11.24
45	Misralb	10.96, 10.96
46	Kirby2	11.24, 10.54, 11.27, 11.37, 10.78
47	Hahn1	11.84, 11.46, 11.07, 11.23, 12.07, 11.29, 11.02
48	Nelson	Two independents, not solvable
49	MGH17	11.09, 11.20, 11.04, 11.35, 10.83
50	Lanczos1	11.35, 10.58, 11.29, 10.94, 10.61, 11.84
51	Lanczos2	12.08, 10.40, 11.25, 11.48, 10.75, 11.50
52	Gauss3	11.88, 10.97, 10.73, 10.81, 11.26, 11.40, 10.48, 10.71
53	Misralc	11.33, 10.80
54	Misrald	12.62, 12.85
55	Roszmzn1	11.84, 10.72, 10.94, 11.68
56	ENSO	10.66, 11.90, 11.11, 11.31, 10.78, 11.40, 12.06, 11.68,
57	MGH09	11.27, 10.52, 10.55, 10.49
58	Thurber	10.44, 10.55, 11.23, 12.80, 11.33, 11.01, 11.43
59	BoxBOD	11.73, 11.83
60	Rat42	11.23, 11.49, 11.82
61	MGH10	11.30, 11.66, 10.92

available from the author (David Heiser) are the VBA modules that have the VBA functions and subroutines for the entire NIST StRD set of equations.

[6] The subroutine LMNoLinearFit should be version 'mod 9.4.2006' or later. Earlier versions are not successful on the NIST StRD data. This version has a Newton-Raphson ending method that improves accuracy by 2-4 LRE units.

NIST Number	NIST Name	Fitted Coefficient LRE Values
62	Eckere4	10.52, 11.25, 11.07
63	Rat43	11.13, 10.53, 10.47, 10.29
64	Bennett5	11.09, 11.12, 11.70

Compare the LRE values with that given in Table 11-5.

To show how well the LM method works, 5 files in *.xls format representing 5 of the non-linear NIST problems are included on the site under the directory, "Sample Test Files". They show how to use the Levenberg-Marquardt method to solve for parameters in non-linear equations. They include derivative functions and a test for the final equation derivative subroutine.

THE NELDER-MEAD DOWNHILL SIMPLEX ROUTINE

The Nelder-Mead Downhill-Simplex, minimizing or maximizing a cell on a worksheet, given a range of cells containing the input values to change, works very well, providing there are no constraints. It does not require analytic derivatives. The limit is 5 fitted parameter values, and the normal run (stopping point) is 300 trials. The Nelder-Mead Downhill-Simplex works well. It is quick, and can be repeated until the objective cell stabilizes to a "no-change" region. LRE values on accuracy, are less than what can be obtained from the Levenberg-Marquardt fit, which requires analytic derivatives.

The Nelder-Mead Downhill Simplex will also provide solutions to multivariate non-linear equations of more than 1 variable. The limit on fitting a maximum of 5 parameters limits its application in a general sense.

Table 11-9: LRE Values From The Nelder-Mead Downhill Simplex

Sequence Number	NIST Name	Fitted Coefficient LRE Values
38	Misrala	8.65,8.56
39	Chwirut2	7.16, 7.462 7.47
40	Chwirut1	7.94, 8.20, 8.50
41	Lanczos3	More than 5 parameters
42	Gauss1	More than 5 parameters
43	Gauss2	More than 5 parameters
44	DanWood	8.84, 8.99
45	Misralb	8.56, 8.51
46	Kirby2	7.51, 8.14, 8.84, 8.30, 8.77
47	Hahn1	More than 5 parameters
48	Nelson	8.76, 6.63, 7.84
49	MGH17	*8.87, 7.61, 7.49, 8.10, 8.06
50	Lanczos1	More than 5 parameters
51	Lanczos2	More than 5 parameters

Sequence Number	NIST Name	Fitted Coefficient LRE Values
52	Gauss3	More than 5 parameters
53	Misralc	8.64, 8.57
54	Misrald	8.61, 8.55
55	Roszman1	7.90, 7.23, 8.21, 7.36
56	ENSO	More than 5 parameters
57	MGH09	*, 8.76, 6.93, 7.23, 7.10
58	Thurber	More than 5 parameters
59	BoxBOD	8.56, 7.81
60	Rat42	9.10, 9.34, 9.02 (first pass)
61	MGH10	*, 7.54, 8.40, 8.62
62	Eckere4	8.98, 9.05, 10.73 (first pass)
63	Rat43	8.88, 7.38, 7.54, 7.31 (first pass)
64	Bennett5	0.53, 1.33, 1.43 (first pass, failed to improve much)

The equations were in the form of VBA functions, which do not have the precedence problem. All cases were started with the NIST case 1 starting value, but some had to be restarted from the second NIST set. * indicates that many passes were made to attain the shown LRE values (i.e. convergence to a “stable” value set).

The process was one of repeating the “300 iteration pass” until it appeared that a stable set of parameter values was obtained. These values were different for each “300 iteration pass”, but appeared to vary within a “stable” interval.. Consequently the reported LRE values are only “neighborhood” values, and should only be viewed in terms of the first 1 or 2 figures.

The MGH series were all a problem on getting a high LRE value. They had to be started from the second starting set, the first set gave LRE values in the –1 to 1 range after repeated reruns. Bennett5 was the only total failure. The problem with Bennett5 was that it jumped to a near solution, but could not improve it. Each 300-pass run improved it by only about a few hundredths of an LRE unit.

In a general sense, we can say that the Nelder-Mead Downhill Simplex will perform better than Solver, but it too has faults.

OTHER ADD-INS FROM THE FOXES TEAM

These are a set of routines for use in Excel that are freely available from the Foxes team (see Note XN). Algorithms included in the Optimiz.xla package are:

Nelder-Mead Downhill-Simplex
Newton-Raphson
Levenberg- Marquardt Least Squares Fitting
Conjugate-Gradient
Davidon-Fletcher-Powell
Random (values)
Divide-Conquer

Parabolic

Some of the routines allow for constraints on variable values to be defined. This gives a general capability for solutions of general operations-research type problems.

NONLINEAR STRUCTURAL EQUATION FITTING

This is a different class of “non-linear equation fitting”. It basically is a method of fitting a covariance or correlation matrix of data to a model. See Bollen 1989. The individual model linear equations with parameters that are to be fitted, can be restructured into an implied covariance/correlation matrix. The difference between them can be reduced to a chi-square statistic. By changing model parameter values, a fit to the correlation/covariance data matrix can be found. The model fit as a chi-square measure then is minimized by changing parameter values. In commercial software, a maximum likelihood method is used to find a parameter value set that minimizes the error.

Excel is not adapted to these methods, because of its limited matrix handling capabilities. Miles (2005) however used Excel and Solver to do a confirmatory factor analysis. Using the first six items of the Surtees and Miller I-GHQ questionnaire responses, Miles (2005) compared the Excel model fits with solutions using Mplus 2.14 and AMOS 5.0 software packages. They were the 6x1 loading matrix (L) and the 6x1 error matrix (E). There was one item (latent variable) in the model.

Table 11-10: Miles L and E Reported Matrix Values

Matrix	Excel	Amos	Mplus	LRE
L-1	.673	.672	.672	2.83
L-2	.549	.548	.548	2.74
L-3	.749	.748	.748	2.87
L-4	.475	.475	.475	16
L-5	.720	.719	.719	2.86
L-6	.742	.741	.741	2.87
E-1	.351	.350	.350	2.54
E-2	.531	.530	.530	2.72
E-3	.244	.244	.244	16
E-4	.507	.505	.505	2.50
E-5	.262	.262	.262	16
E-6	.274	.273	.273	2.44

The LRE values are biased, since only 3 decimal figures were reported for parameter values. These results are consistent with the accuracies of Solver as reported in table 11-4.

His conclusions were, “The spreadsheet has little functionality and would not be recommended as a useful tool; it does provide a useful pedagogical exercise.” He does say that the functionality can be added to, and vba can be used to automate cell equations and function inputs.

He (Miles 2007) also commented “I'm basically using Excel because I can write a simple program in it, which can loop and write output files. Given reasonable starting values, I've always found it to converge on the same solution as an SEM program.”

This all goes back to ground zero, what is the accuracy required and what is the users acceptance of the numbers as being “true”? Belief in a computer output is a fundamental human characteristic.

THE BOTTOM LINE

There is no one universal method for fitting non-linear equations to data when analytic derivatives cannot be calculated.

Outside of methods that transform the data and equation to a linear model and then use LINEST to find parameter values; all other methods are indirect approximations to equation coefficient values.

The only reliable method of fitting non-linear equations to data is a method that uses analytical derivatives.

All the indirect approximation methods require a starting set of parameter values, which must be in the region of a global minimum of a fit measure. Otherwise one is likely to end up with an incorrect parameter value set.

The use of a mean-error-square (RMSE) method of fit is a common method, but is not a universal criteria of fit. Other methods can be used, which will give different parameter values.