Classroom Tips and Techniques: Nonlinear Fit, Optimization, and the *DirectSearch* Package

Robert J. Lopez
Emeritus Professor of Mathematics and Maple Fellow
Maplesoft

Introduction

In this month's article, I revisit a nonlinear curve-fitting problem that appears in my Advanced Engineering Mathematics ebook, examine the role of Maple's Optimization package in that problem, and then explore the *DirectSearch* package from Dr. Sergey N. Moiseev.

The nonlinear curve-fitting problem was brought to me by one of the statisticians at the Rose-Hulman Institute of Technology well before my 2003 retirement from the classroom. Originating in the newly established biomed faculty at RHIT, its experimentally obtained data points were to be fit by the Michaelis-Menton function $f(s) = a\,s/(b+s)$.

When I initially examined the problem in Maple V Release 4, I found that the **fit** command in the (now deprecated) *stats* package only applied to linear models. Hence, when this problem became a section in the 2001 Addison-Wesley version of my Advanced Engineering Math text, it was solved from first principles by forming and minimizing SS , the sum of squares of deviations. Since the text predated the *Optimization* package, SS was minimized by techniques of the calculus. (The essence of the discussion in the text is a comparison between the nonlinear fit and two different linearizations. The linearizations prove to be poor fits!)

By the time the AEM ebook was prepared in Maple 10, the *Optimization* package had been added to Maple, and its use for this problem was then included in the ebook. The results for the minimization of SS , as one would expect, agree for the two approaches. Moreover, the **NonlinearFit** command in the *Statistics* package (new in Maple 10) provided an immediate fit of the Michaelis-Menton model to the given data.

The recent announcement in MaplePrimes of version 2 of the *DirectSearch* package caught my attention. I was curious to see to what extent this package would have been of use for the curve-fitting problem in the AEM ebook. I was surprised (and delighted) to discover how robust this package actually is.

Michaelis-Menton Fit to Biomed Data

Figure 1 is a graph of the 46 data points that are to be fit with the Michaelis-Menton function $f(s) = a\,s/(b+s)$. The **plot** command and the actual data are "behind" the table containing the graph. To see them, select the "Show input" option in the Table Properties dialog. The data, $S = [s_1,\ldots, s_{46}],\ V = [v_1,\ldots, v_{46}]$, are given in floating-point form, to six places.

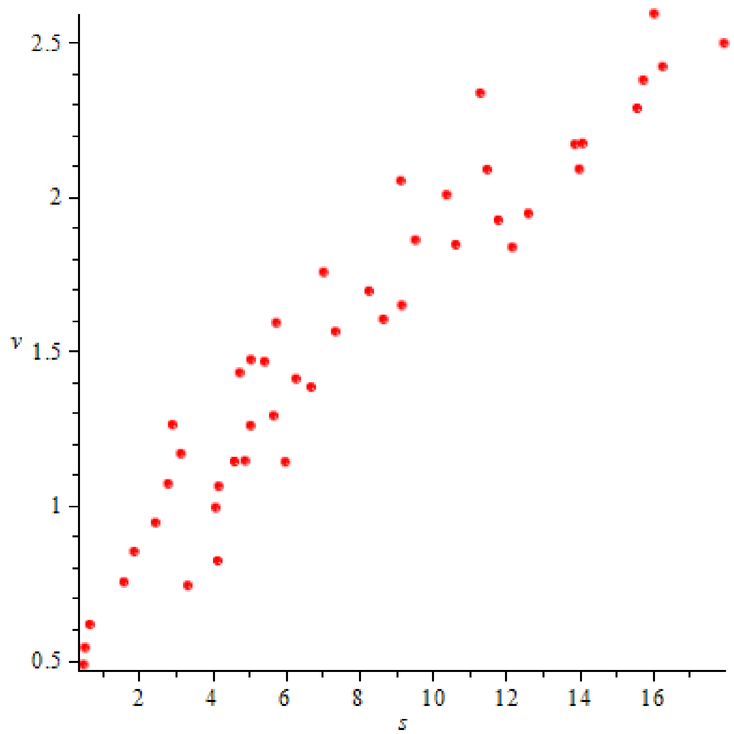

Figure 1 Forty-six data points (s_k, v_k) to be fit with $f(s) = \frac{a\,s}{b+s}$

The following definition of the Michaelis-Menton function also takes place in a table for which the input is not shown.

$$f(s) = \frac{a\,s}{b+s}$$

Prior to Maple 10, there was no direct functionality for a nonlinear fit in Maple. Hence, from first principles, we let SS be the sum of squares of deviations for $f(s)$ and the given data:

$$SS := add\left(\left(f\left(S_k\right) - V_k\right)^2, k = 1..46\right):$$

Figure 2 is a graph of a portion of the surface $SS(a, b)$. It shows the existence of a minimum point, and also shows that this minimum occurs in a steep, but narrow, "valley." Again, the graph appears in a table that conceals input.

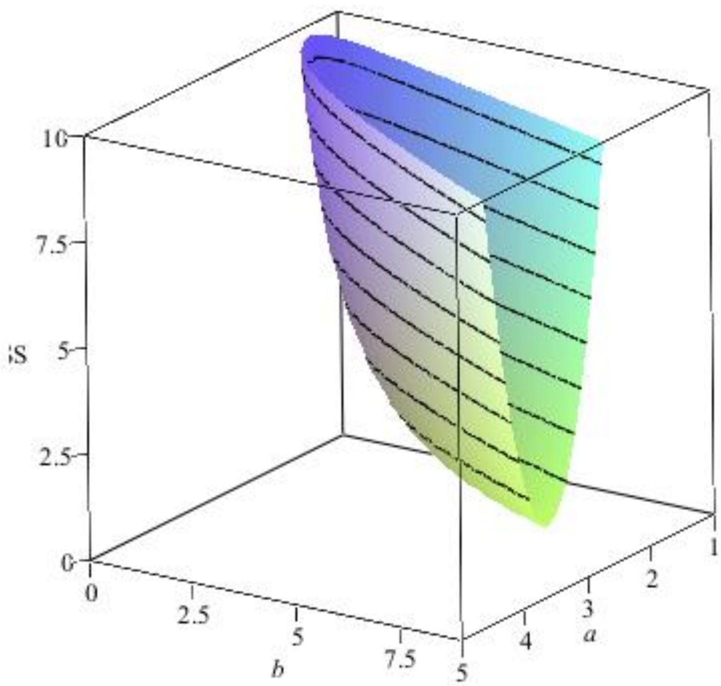

Figure 2 The sum of squares of the deviations graphed as a function of the parameters a and b

The ordinary tools of the calculus provide the minimizing values of a and b,

$$fsolve\left(\left\{\frac{\partial}{\partial a} SS = 0, \frac{\partial}{\partial b} SS = 0\right\}, \{a, b\}, a = 1..5, b = 5..10\right)$$

$$\{a = 3.477708271, b = 8.224627001\}$$

and hence of the fitting function itself

$$F := eval(f(s), \mathbf{(1)}) \quad = \quad \frac{3.477708271\, s}{8.224627001 + s}$$

and the minimum value of SS, namely,

$$SSF := eval(SS, \mathbf{(1)}) \quad = \quad 1.622578612$$

Applying the *Optimization* package's **Minimize** command to SS gives

$$Optimization\text{:-}Minimize(SS)$$

$$[1.62257861139101190, [a = 3.47770700808554, b = 8.22462128995005]]$$

This extreme value is not as accurate as we obtained in . An improvement can be obtained with the *Optimization* package's **NLPSolve** command, which became available in Maple 9.5.

$$Optimization\text{:-}NLPSolve(SS, method = nonlinearsimplex, evaluationlimit = 200, optimalitytolerance = 1\mathrm{e}{-15})$$

$$[1.62257861139007753, [a = 3.47770826399059, b = 8.22462700440175]]$$

From Maple 10 onwards, the **NonlinearFit** command from the *Statistics* package immediately gives

$$Statistics\text{:-}NonlinearFit(f(s), S, V, s)$$

$$\frac{3.47770837575172\, s}{8.22462695881459 + s}$$

The *DirectSearch* Package

Description

Version 2 of the *DirectSearch* package is available from the Maple Application Center. Installation instructions are provided in the README.txt file that accompanies the code. Although written for Maple 13, I've used it in Maple 14 and Maple 15 with no ill effects.

The code for the commands in the package are in the archive DirectSearch.mla; the archive DirectSearch.hdb contains the files for the help pages, which are very good. There are three different optimizers: Search, GlobalOptima, and Global Search, all of which use derivative-free direct search methods so all the computations in the package apply to functions that may not be differentiable or continuous. In contrast, the only derivative-free algorithm in Maple's *Optimization* package is the sequential simplex technique of Nelder and Mead, implemented by the nonlinearsimplex option in the **NLPSolve** command.

For multiobjective optimization, the DirectSearch package has the seven commands **Minimax**, **CompromiseProgramming**, **WeightedSum**, **BoundedObjective**, **ModifiedTchebycheff**, **WeightedProduct**, and **ExponentialWeightedSum**. In addition, there is the **SolveEquations** command for solving systems of equations, and the **DataFit** command for nonlinear fits to data. The **DataFit** command supports nine different fitting methods, best described via Table 1, extracted from the help page for this command.

Fit method	Method description	Objective function
lms	Least mean squares method. This method is effective but not robust to outliers.	$F(\mathbf{a}) = \frac{1}{k}\sum_{i=1}^{k} w_i \cdot \left[Y_i - f\left(\mathbf{X}_i; \mathbf{a}\right) \right]^2$
lad	Least absolute deviations method. This method is more robust to outliers than LMS method.	$F(\mathbf{a}) = \frac{1}{k}\sum_{i=1}^{k} w_i \cdot \left\| Y_i - f\left(\mathbf{X}_i; \mathbf{a}\right) \right\|$
lts	Least trimmed squares method. Robustness of this method depends on upper percentile value *p* ($p = 95$ by default).	$F(\mathbf{a}) = \mathrm{TrimmedMean}(\mathbf{r}, 0, p)$, $r_i = w_i \cdot \left\| Y_i - f\left(\mathbf{X}_i; \mathbf{a}\right) \right\|^2,\ i = 1 .. k$, where TrimmedMean is trimmed mean
lws	Least Winsorized squares method. Robustness of this method depends on upper percentile value *p* ($p = 95$ by default).	$F(\mathbf{a}) = \mathrm{WinsorizedMean}(\mathbf{r}, 0, p)$, $r_i = w_i \cdot \left\| Y_i - f\left(\mathbf{X}_i; \mathbf{a}\right) \right\|^2,\ i = 1 .. k$, where WinsorizedMean is Winsorized mean
minimax	Minimax method. This method is very sensitive to outliers.	$F(\mathbf{a}) = \max_i \left(w_i \cdot \left\| Y_i - f\left(\mathbf{X}_i; \mathbf{a}\right) \right\| \right),\ i = 1 .. k$
median	Least median squares method. This method is most robust to outliers.	$F(\mathbf{a}) = \mathrm{me}\left(w_i \cdot \left\| Y_i - f\left(\mathbf{X}_i; \mathbf{a}\right) \right\|^2 \right),\ i = 1 .. k$, where $\mathrm{me}(\ )$ is sample median
quantile	Least quantile squares method. Robustness of this method depends on quantile order *p* ($p = 0.75$ by default).	$F(\mathbf{a}) = \mathrm{q}_p\left(w_i \cdot \left\| Y_i - f\left(\mathbf{X}_i; \mathbf{a}\right) \right\|^2 \right),\ i = 1 .. k$, where $\mathrm{q}_p(\ )$ is sample quantile of order p
trimean	Trimean method. This method is robust to outliers.	$F(\mathbf{a}) = \frac{1}{2}\left(\mathrm{me}\left(r_i\right) + \frac{q_{0.25}\left(r_i\right) + q_{0.75}\left(r_i\right)}{2} \right)$, $r_i = w_i \cdot \left\| Y_i - f\left(\mathbf{X}_i; \mathbf{a}\right) \right\|,\ i = 1 .. k$, where $\mathrm{me}(\ ), \mathrm{q}_p(\ )$ is sample median and quantile of order *p*

lfad	Least function of absolute deviations method. Robustness of this method depends on the function g specified (by default, $g(x) = \ln(1+x)$).	$F(\mathbf{a}) = \frac{1}{k}\sum_{i=1}^{k} w_i \cdot g\left(\left\|Y_i - f\left(\mathbf{X}_i; \mathbf{a}\right)\right\|\right)$

Table 1 Options to the **DataFit** command in the *DirectSearch* package

Application 1

Our first application of *DirectSearch* is to the problem iconized by Figure 1. According to Table 1, the "ordinary" least squares method is implemented in *DirectSearch* as

$$DirectSearch:\text{-}DataFit(f(s), Vector(S), Vector(V), s, fitmethod = lms)$$

$$[0.0352734480736973, [a = 3.47770827104531, b = 8.22462699897286], 96]$$

The first number is the minimum value of the objective function $F(\mathbf{a})$ defined in the top row of Table 1. Note that this function differs from our SS by the factor $1/k$, where, for this calculation, $k = 46$. Hence, we compute

$$46 \cdot 0.0352734480736973 = 1.622578611$$

results compatible with those obtained earlier.

Application 2

Exercise 3 in Section 47.3 of my Advanced Engineering Mathematics ebook is the isoperimetric problem asking for the extremal $y(x)$ that renders a functional J stationary, satisfies $y(0) = 1$, has arc length $3/2$, and intersects the transversal $y(x) = x^2$, where J is given by

$$J = \int_0^{\beta} (x + y)\, dx$$

An extremal $y(x)$, an arc of the circle $(x-a)^2 + (y-b)^2 = a^2 + (1-b)^2$, is determined by a triple (a, b, β) that is a solution of the following three equations.

$$g_1 := \sqrt{a^2+(1-b)^2}\left(\arcsin\left(\frac{\beta-a}{\sqrt{a^2+(1-b)^2}}\right)\right.$$
$$\left.+\arcsin\left(\frac{a}{\sqrt{a^2+(1-b)^2}}\right)\right)-\frac{3}{2}=0:$$
$$g_2 := \beta^4-\beta^2 b+\beta^2-2\beta a-2a\beta^3+2a\beta b-\beta^3+\beta b-1+2b$$
$$-b^2=0:$$
$$g_3 := (\beta-a)^2+\left(\beta^2-b\right)^2-a^2-(1-b)^2=0:$$

Maple's **fsolve** command (numeric solver) finds one solution, namely,

$$fsolve\left(\{g_1, g_2, g_3\}, \{a=.3, b=2, \beta=1.2\}\right)$$
$$\{a=0.2998750339, b=2.003187464, \beta=1.252187766\}$$

Unfortunately, it fails to find any other solution. There is a second solution, found by solving the middle equation for a, and so eliminating a from the first and third equations. This leads to two equations of the form $G(b,\beta)=0$, whose graph reveals that there are two intersections of the two implicit functions so defined. The second solution is then found to be

$$\{a=0.4888015056, b=0.9590573224, \beta=0.9793147208\}$$

The **SolveEquations** command in *DirectSearch* seeks minima of $R=\sum_{k=1}^{n} g_k^2$, and returns an $m\times 4$ matrix, each row of which details an extreme point: the value of R, a vector of residuals for each g_k, the coordinates of the point, and the number of evaluations of R it took to find that point. The minima are arranged in order of increasing value of R.
With n = 3, our equations yield the matrix

$$M := DirectSearch\text{:-}SolveEquations\left([g_1, g_2, g_3], AllSolutions,\right.$$
$$\left. evaluationlimit=1000000\right):$$

which has some 83 rows. The first two solutions are

$$M[1,3]$$
$$[a=0.299875033923247, b=2.00318746378598, \beta$$
$$=1.25218776613643]$$

and

$$M[2,3]$$
$$[a=0.489458553450717, b=0.961264302816779, \beta$$
$$=0.980433456998931]$$

for which the R-values are on the order of 10^{-20} and 10^{-10}, respectively. The value of Rin the third row is 1.7×10^{-5}; in the last row it is greater than 7. Hence, we conclude that there are just two solutions of our three equations and these solutions correspond to the two found in Maple.

Application 3

Some time ago I was asked if Maple could solve the following set of five equations for $\alpha, w_k, \rho_k, k = 1, 2$.

$$e_1 := 0.0003073722490 = \frac{1}{2}w_1 + \frac{1}{2}w_2 + \frac{1}{2}\sqrt{(w_1 - w_2)^2 \cosh(2\alpha)^2 + (\rho_1 - \rho_2)^2 \sinh(2\alpha)^2} :$$

$$e_2 := 0.0006545060363 = \frac{1}{2}w_1 + \frac{1}{2}w_2 - \frac{1}{2}\sqrt{(w_1 - w_2)^2 \cosh(2\alpha)^2 + (\rho_1 - \rho_2)^2 \sinh(2\alpha)^2} :$$

$$e_3 := 0.001043680000 = \frac{1}{2}\rho_1 + \frac{1}{2}\rho_2 + \frac{1}{2}\sqrt{(\rho_1 - \rho_2)^2 \cosh(2\alpha)^2 + (w_1 - w_2)^2 \sinh(2\alpha)^2} :$$

$$e_4 := 0.0003157290000 = \frac{1}{2}\rho_1 + \frac{1}{2}\rho_2 - \frac{1}{2}\sqrt{(\rho_1 - \rho_2)^2 \cosh(2\alpha)^2 + (w_1 - w_2)^2 \sinh(2\alpha)^2} :$$

$$e_5 := 0.5655930794 = \sinh(2\alpha)\cosh(2\alpha)\left(\frac{\rho_2 - \rho_1}{w_1 - w_2} + \frac{w_1 - w_2}{\rho_2 - \rho_1}\right) :$$

Maple's **fsolve** command does not provide a solution, and its **LSSolve** command from the *Optimization*package gives different solutions (depending on the initial point), generally with sum of squares approximately 3×10^{-8} . I never could bring my self to believe that any of these minima for the sum of squares was actually a solution of the equations.

Of course, when I started exploring *DirectSearch*, I tried

$$N := DirectSearch\text{:-}SolveEquations([seq(e[k], k = 1..5)], AllSolutions, evaluationlimit = 1000000) :$$

which results in a 22×4 matrix of extrema. The first such extremum, namely,

$evalf(N[1,3],5)$

$$\left[\alpha = -0.031530, \rho_1 = 0.0010495, \rho_2 = 0.00033972, w_1 = 0.00050178, w_2 = 0.00042141\right]$$

has a residual of 9.8×10^{-8}, but no solution produced by **SolveEquations** matches any returned by **LSSolve**. The existence of a solution now becomes an urgent issue.

To this end, consider the difference

$$e_1 - e_2$$

$$-0.0003471337873 = \sqrt{(w_1 - w_2)^2 \cosh(2\alpha)^2 + (\rho_1 - \rho_2)^2 \sinh(2\alpha)^2}$$

It would appear from the difference in signs on the left and right sides of that no solution exists. Hence, we must be careful with numeric "solutions" obtained by minimizing $R = \sum_{k=1}^{n} g_k^2$. Just because a powerful minimization tool is used does not necessarily mean that a solution has been found for the set of equations $g_k, k = 1,\ldots,n$.