ELSEVIER

Computer Methods and Programs in Biomedicine 65 (2001) 191–200

Computer Methods and Programs in Biomedicine

www.elsevier.com/locate/cmpb

A step-by-step guide to non-linear regression analysis of experimental data using a Microsoft Excel spreadsheet

Angus M. Brown *

Department of Neurology, Box 356465, University of Washington School of Medicine, Seattle, WA 98195-6465, USA

Received 20 February 2000; received in revised form 8 May 2000; accepted 20 June 2000

Abstract

The objective of this present study was to introduce a simple, easily understood method for carrying out non-linear regression analysis based on user input functions. While it is relatively straightforward to fit data with simple functions such as linear or logarithmic functions, fitting data with more complicated non-linear functions is more difficult. Commercial specialist programmes are available that will carry out this analysis, but these programmes are expensive and are not intuitive to learn. An alternative method described here is to use the SOLVER function of the ubiquitous spreadsheet programme Microsoft Excel, which employs an iterative least squares fitting routine to produce the optimal goodness of fit between data and function. The intent of this paper is to lead the reader through an easily understood step-by-step guide to implementing this method, which can be applied to any function in the form $y=f(x)$, and is well suited to fast, reliable analysis of data in all fields of biology.

Keywords: Microsoft Excel; Non-linear regression; Least squares; Iteration; Goodness of fit; Curve fit

1. Introduction

The use of curve fitting to describe experimental data is widespread in all fields of biology. The purpose of such analysis is to standardize data interpretation into a uniformly recognized form. Curve fitting essentially describes the experimental data as a mathematical equation in the form $y=f(x)$, where x is the 'independent' variable and is controlled by the experimenter; y is the 'dependent' variable, which is measured; and f is the function, which includes one or more parameters used to describe the data. The better the fit, the more accurately the function describes the data. The introduction of personal computers into laboratories has greatly reduced the time and effort required in analyzing data and it is a relatively straightforward process to fit data with simple functions such a linear regression, a process that can be carried out with a few simple point-and-click commands. It is more difficult, however, to fit data with more complicated non-linear functions. This is usually carried out by specialist programmes such as Microcal Origin, Sigma Plot

* Tel.: +1-206-6168278; fax: +1-206-6858100.
E-mail address: ambrown@u.washington.edu (A.M. Brown).

0169-2607/01/$ - see front matter
PII: S0169-2607(00)00124-3

or Graphpad Prism. An advantage of these programmes is that they are capable of fitting user-input functions to data. However these programmes tend to be expensive (in the £500 range), and if the goal is simply to fit data with a non-linear function, the user pays for a vast excess of redundant features. These programmes are aimed at experienced specialist users with a mathematical background and tend to be difficult for the novice to learn. Additionally, these programmes do not handle data manipulation well and tend to display data, graphs, results, and analysis in a multitude of separate windows, which can lead to confusion.

An alternative method is to use Microsoft Excel to fit non-linear functions. An advantage of this method is that Excel is probably included in the computer package as part of Microsoft Office, and thus no additional expense is required. Spreadsheet programmes are among the most commonly used software, and most biologists have experience with them even if at an elementary level. Excel offers a friendly user interface, flexible data manipulation, built-in mathematical functions and instantaneous graphing of data. Excel contains the SOLVER function, which is ideally suited to fitting data with non-linear functions via an iterative algorithm [1], which minimizes the sum of the squared difference between data points and the function describing the data. The objective of this present study was to describe a method of non-linear regression using the SOLVER function of Excel.

2. Method

The method described in this paper, to conduct a curve fitting protocol in an Excel spreadsheet, was carried out on a Gateway Pentium II computer running Microsoft Windows 98 and Excel 97. The protocol involves entering data manually into the spreadsheet and graphing the data. Once the data have been entered, the curve fitting protocol is carried out and the curve fit is overlaid on the data points. Goodness of fit data are also calculated so that the accuracy of fit can be assessed.

2.1. Least squares fit

As a first step to analyzing data using a curve fitting protocol it is necessary to determine the goodness of fit. Essentially this means estimating how well the curve (i.e. the function) describes the data. The most commonly used measure of the goodness of fit is least squares. This is based on the principal that the magnitude of the difference between the data points and the curve is a good measure of how well the curve fits the data. For our purposes the least squares fit method will be illustrated by fitting data with a linear function, a process called linear regression. It is assumed that the reader knows how to input data into an Excel spreadsheet and graph the data, and that readers of this paper will be analyzing data in the form (x, y), where x is the 'independent' variable and y is the 'dependent' variable. The data are input onto the spreadsheet in the form of two columns, one each for the x and y variables. The data are then graphed; the most convenient type of graph for illustrative purposes is a scatter graph. To fit a linear function to the data, highlight one of the data points on the graph by clicking the right-hand mouse button and select Add Trendline. The Add Trendline Dialogue box appears. Highlight the Linear box, and a linear fit is superimposed on the data. The parameters of the fit can be displayed on the graph by highlighting the Option tab in the Add Trendline Dialogue box and selecting Display equation on chart. This process however, does not explain to the user how the fit was determined. The difference between the data and the fit is illustrated by the vertical arrows at each data point in Fig. 1A. The difference, or residual, between each data point and the fit is calculated. This is illustrated in Fig. 1B, where the y value of each point is replaced by the distance of that point from the linear function. The least squares fitting method squares the residual value to eliminate the effects of positive or negative deviation from the fit and is illustrated in Fig. 1C. This is described by:

$$SS = \sum_{i=1}^{n} [y - y_{fit}]^2 \quad (1)$$

where y is the data point, y_{fit} is the value of the curve at point y, and SS is the sum of the squares. The best fit of the data is the linear function that has the smallest value for the squared sum (SS) of all the differences. A linear function is described by:

$$y = mx + c \tag{2}$$

where y is the 'dependent' variable and x is the 'independent' variable. The parameter values m (the slope) and c (the intercept) are calculated by:

$$m = \frac{n\left(\sum xy\right) - \left(\sum x\right)\left(\sum y\right)}{n\left(\sum (x^2)\right) - \left(\sum x\right)^2} \tag{3}$$

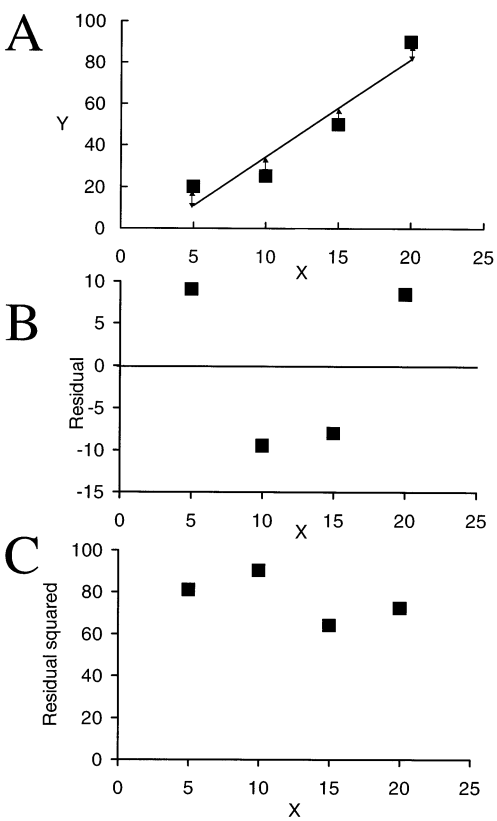

Fig. 1. Linear regression. A: An X-Y Scatter plot illustrating the difference between the data points and the linear fit. B: A residual plot illustrating the difference between data points and the fit. C: The residual is squared to eliminate the effect of positive or negative deviations from the fit. This value is used to calculate the sum of the squares.

and

$$c = \frac{\left(\sum y\right)\left(\sum (x^2)\right) - \left(\sum x\right)\left(\sum xy\right)}{n\left(\sum (x^2)\right) - \left(\sum x\right)^2} \tag{4}$$

respectively. The r^2 value, also known as the correlation index or coefficient of determination, is a value between 0 and 1. It expresses the proportion of variance in the 'dependent' variable explained by the 'independent' variable. An r^2 value of 0 means that knowing x does not help to predict y. As the r^2 value increases towards 1 the more accurately the function fits the data. (N.B. By convention in linear regression the r^2 value is expressed in lower case and in non-linear regression the R^2 value is expressed in upper case).

$$r^2 = 1 - \frac{\sum (y - y_{mean})^2}{\sum (y^2) - \dfrac{\sum (y)^2}{n}} \tag{5}$$

where y is the data point, and y_{mean} is the average value of the y data. This method of least squares fitting can be used only with data in which the 'dependent' variable is a linear function of the 'independent' variable.

2.2. Non-linear regression

Prior to the advent of personal computers and specialist curve fitting programmes non-linear data would be transformed into a linear form and subsequently analyzed by linear regression (e.g. Lineweaver Burke method or Scatchard plots). These transformations could yield inaccurate analysis as the linear regression was carried out on transformed data, which may distort the experimental error or alter the relationship between the x and y values. This method is outdated and inaccurate and should not be used. Instead for data that are not described by a linear function, it is necessary to implement a protocol that will fit a non-linear function to the data. A method that is suitable for this procedure is called iterative non-linear least squares fitting. This process uses the same goal as described for linear regression, i.e.

minimize the value of the squared sum of the difference between data and fit. However it differs from linear regression in that it is an iterative, or cyclical process. This involves making an initial estimate of the parameter values. The initial parameter estimates should be based on prior experience of the data or a sensible guess based on knowledge of the function used to fit the data. The first iteration involves computing the SS based on the initial parameter values. The second iteration involves changing the parameter values by a small amount and recalculating the SS. This process is repeated many times to ensure that changes in the parameter values result in the smallest possible value of SS. For linear regression only a single calculation is required to provide the lowest value of the SS, because the second and higher derivatives of the function are zero. Therefore, the algorithm requires only a single iteration. However, for non-linear regression the second and higher derivatives are not zero, and thus an iterative process is required to calculate the optimal parameter values. Several different algorithms can be used in non-linear regression including the Gauss–Newton, the Marquardt–Levenberg, the Nelder–Mead and the steepest descent methods [2]. SOLVER, however, uses another iteration protocol, which is based on the robust and reliable generalized reduced gradient (GRG) method. A detailed description of the evolution and implementation of this code can be found elsewhere [3,4]. All of these algorithms have similar properties. They all require the user to input initial parameter values and use these values to provide a better estimate of the parameters employing an iterative process. With the same set of data all of these methods should yield the same parameter values.

The following example illustrates how to use the SOLVER function in Excel to fit data with user-input non-linear functions. The process by which the curve fit proceeds is called iterative non-linear least squares regression. The example used is a sigmoidal function (the Boltzmann equation), which describes the probability that an ion channel will be open relative to voltage. This example is used purely for illustrative purposes and it is not necessary that the reader understand anything about ion channels.

The Boltzmann function is described by the following function:

$$y = \frac{1}{\left[1 + \exp\frac{(V - E)}{\text{Slope}}\right]} \tag{6}$$

where y is the 'dependent' variable, E is the 'independent' variable (Voltage), and V and Slope are the parameter values. V is the half activation voltage, which describes the voltage at which half of the ion channels are open (i.e. where $y = 0.5$). Slope describes the slope at the point V and indicates the steepness of the curve, or sensitivity to voltage of the ion channel.

2.3. Configuring the spreadsheet for non-linear regression

In order to perform non-linear regression analysis using the Boltzmann function, the following procedure must be carried out:

1. Input onto a spreadsheet the raw data in two columns, the X column containing the 'independent' variable (Voltage), and the Y column containing the 'dependent' variable (Data). This is illustrated as Columns A and B (Voltage and Data, respectively) of Fig. 2A, where Voltage is the 'independent' variable and Data is the 'dependent' variable.

2. Graph the data contained in cells A2 to B20 in a Scatter plot. The data points are displayed as filled squares.

3. Enter labels in cells G1 to G8 to describe the contents of the adjacent cells. In cell G1 enter V, which will describe the parameter in cell H1. For cell H1 select the Insert menu choose Name then Define for cell H1. Name the cell V. Similarly, for cells G2 to G8, enter Slope, Mean of y, df, S.E. of y, R2. Critical t and CI, respectively. Name cells H2 to H8, Slope, Mean_of_y, df, S.E._of_y, RSQ, Critical_t and CI, respectively.

4. In Column C (Boltzmann) enter the equation describing the Boltzmann function. This has been rearranged from Eq. (6) into a form that Excel recognizes:

$$= (1/(1 + \text{EXP}((V - \text{A2})/\text{Slope})))$$

A

	A	B	C	D	E	F	G	H
1	Voltage	Data	Boltzmann	Upper CI	Lower CI		V	-20
2	-60	0	=(1/(1+EXP((V-A2)/Slope)))	=C2+CI	=C2-CI		Slope	10
3	-55	0	=(1/(1+EXP((V-A3)/Slope)))	=C3+CI	=C3-CI		Mean_of_y	=AVERAGE(B2:B20)
4	-50	0.05	=(1/(1+EXP((V-A4)/Slope)))	=C4+CI	=C4-CI		df	=COUNT(B2:B20)-COUNT(H1:H2)
5	-45	0.08	=(1/(1+EXP((V-A5)/Slope)))	=C5+CI	=C5-CI		SE of y	=SQRT(SUM((B2:B20-C2:C20)^2)/df)
6	-40	0.1	=(1/(1+EXP((V-A6)/Slope)))	=C6+CI	=C6-CI		R^2	=1-SUM((B2:B20-C2:C20)^2)/SUM((B2:B20-Mean_of_y)^2)
7	-35	0.15	=(1/(1+EXP((V-A7)/Slope)))	=C7+CI	=C7-CI		Critical t	=TINV(0.05,df)
8	-30	0.18	=(1/(1+EXP((V-A8)/Slope)))	=C8+CI	=C8-CI		CI	=Critical_t*SE_of_y
9	-25	0.2	=(1/(1+EXP((V-A9)/Slope)))	=C9+CI	=C9-CI			
10	-20	0.3	=(1/(1+EXP((V-A10)/Slope)))	=C10+CI	=C10-CI			
11	-15	0.4	=(1/(1+EXP((V-A11)/Slope)))	=C11+CI	=C11-CI			
12	-10	0.5	=(1/(1+EXP((V-A12)/Slope)))	=C12+CI	=C12-CI			
13	-5	0.6	=(1/(1+EXP((V-A13)/Slope)))	=C13+CI	=C13-CI			
14	0	0.7	=(1/(1+EXP((V-A14)/Slope)))	=C14+CI	=C14-CI			
15	5	0.8	=(1/(1+EXP((V-A15)/Slope)))	=C15+CI	=C15-CI			
16	10	0.85	=(1/(1+EXP((V-A16)/Slope)))	=C16+CI	=C16-CI			
17	15	0.89	=(1/(1+EXP((V-A17)/Slope)))	=C17+CI	=C17-CI			
18	20	0.9	=(1/(1+EXP((V-A18)/Slope)))	=C18+CI	=C18-CI			
19	25	0.95	=(1/(1+EXP((V-A19)/Slope)))	=C19+CI	=C19-CI			
20	30	1	=(1/(1+EXP((V-A20)/Slope)))	=C20+CI	=C20-CI			

B

	A	B	C	D	E	F	G	H
1	Voltage	Data	Boltzmann	Upper CI	Lower CI		V	-10.317
2	-60	0	0.017	0.059	-0.026		Slope	12.194
3	-55	0	0.025	0.068	-0.018		Mean of y	0.455
4	-50	0.05	0.037	0.080	-0.005		df	17
5	-45	0.08	0.055	0.098	0.012		SE of y	0.020
6	-40	0.1	0.081	0.123	0.038		R^2	0.997
7	-35	0.15	0.117	0.159	0.074		Critical t	2.110
8	-30	0.18	0.166	0.209	0.123		CI	0.043
9	-25	0.2	0.231	0.273	0.188			
10	-20	0.3	0.311	0.354	0.269			
11	-15	0.4	0.405	0.448	0.363			
12	-10	0.5	0.506	0.549	0.464			
13	-5	0.6	0.607	0.650	0.565			
14	0	0.7	0.700	0.742	0.657			
15	5	0.8	0.778	0.821	0.736			
16	10	0.85	0.841	0.884	0.799			
17	15	0.89	0.889	0.931	0.846			
18	20	0.9	0.923	0.966	0.881			
19	25	0.95	0.948	0.990	0.905			
20	30	1	0.965	1.007	0.922			

Fig. 2. Spreadsheet template for non-linear regression. A: The data are entered into Column A and B with Column C used to generate the fit based on the parameters in Cells H1 and H2. Columns D and E calculate the 95% confidence interval around the fit. Cell H6 is used to calculate R^2. B: The solution of the fit calculated by SOLVER.

where V and Slope refer to the parameter values in cells H1 and H2.

5. Copy the equation from cell C2 down to and including C20. Note that A2 is a 'relative reference', which specifies the location of a cell relative to the cell in which the calculation will be carried out, in this case cell C2. Thus copying from Rows 2 to 20, changes the value of A2 to reflect the appropriate Row.

6. The mean of the y values is calculated by entering the following formula in H3.

$$= \text{AVERAGE(B2:B20)}$$

7. The degrees of freedom is defined as the number of data point minus the number of parameters in the function. It is calculated by entering the following formula in H4.

$$= \text{COUNT(B2:B20)} - \text{COUNT(H1:H2)}$$

8. The standard error of the y values is defined as

$$\text{S.E.} = \sqrt{\frac{\sum(y - y_{\text{fit}})^2}{\text{df}}} \quad (7)$$

and is calculated by entering the following formula in H5

$$= \text{SQRT(SUM((B2:B20} - \text{C2:C20)} \wedge 2)/\text{df}).$$

However as this formula must be expressed as an array formula, press Ctrl + Shift + Enter. This encloses the whole formula within a pair of curly brackets ({}), denoting it as an array formula.

9. The R^2 value, the correlation index or coefficient of determination, is defined as

$$R^2 = 1 - \frac{\sum(y - y_{\text{fit}})^2}{\sum(y - y_{\text{mean}})^2} \quad (8)$$

and is calculated by entering the following formula in H6 and expressing it as an array formula as described above

$$= 1 - \text{SUM((B2:B20} - \text{C2:C20)} \wedge 2)$$
$$/\text{SUM((B2:B20} - \text{Mean\_of\_}y) \wedge 2)$$

10. In order for the confidence interval of the fit to be calculated the critical t value at a significance level of 95% is calculated by entering the following formula in H7.

$$= \text{tinv(0.05,df)}$$

The confidence interval is defined as

$$y_{\text{fit}} * \text{ Critical\_}t\text{*S.E.\_of\_}y$$

Thus in H8 enter

$$= \text{Critical\_}t\text{*S.E.\_of\_}y$$

Enter the following formula in D2

$$= \text{C2} + \text{CI}$$

and copy it down to D20. Similarly enter = C2 − CI in E2 and copy down to E20. This calculates the upper and lower confidence limits (95%) of the fit.

11. The S.E. of the y values, R^2 and CI are automatically calculated: 0.134, 0.872 and 0.283, respectively.

12. Insert initial estimates of the parameters V and Slope into cells G1 and G2, respectively. Approximate estimates are −20 and 10, respectively. Fig. 2A illustrates the spreadsheet template with the formulas used in the fitting protocol displayed.

13. Graph Columns C, D and E versus Column A such that they are displayed as continuous lines on the graph as illustrated in Fig. 3A. It can be seen that the initial estimate (thick line) is not a good fit of the data with large confidence limits (thin lines). The following section describes manipulations that allow SOLVER to improve the fit.

2.4. Implementation of SOLVER

The above protocol sets up the spreadsheet template that SOLVER requires in order to fit a curve to the data. This method can be used to fit data with any user input non-linear function. Simply enter the appropriate parameter values in Column H and the function in a form that Excel recognizes in Column C. Carry out the following:

14. Open the SOLVER function, which can be found under the Tools menu. The Dialogue box illustrated in Fig. 4A appears. If SOLVER is not in this menu it should be installed. See Excel documentation for installation procedure.

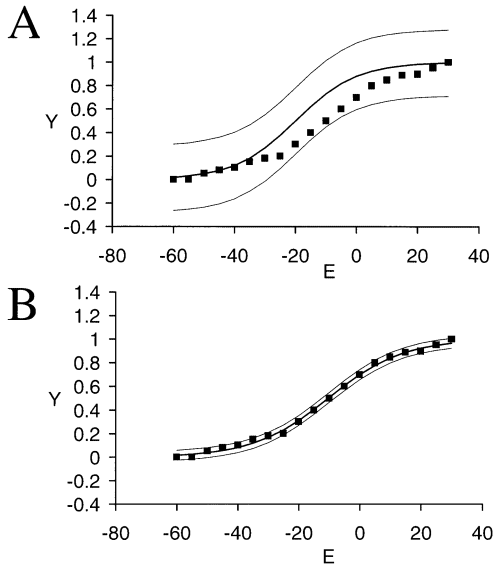

Fig. 3. Boltzmann fit of electrophysiological data. A: This graph displays the experimental data points (filled squares), the fit based on the initial parameter estimates (thick line), and the 95% confidence intervals (thin lines) around the fit. B: The fit as calculated by SOLVER. Note how the fit more accurately overlies the data than the initial estimates, and the CI are closer to the fit.

15. In Set Target Cell box enter RSQ
16. Set the Equal To option to Max. SOLVER tries to maximize the value of R^2.
17. In By Changing Cells box enter V, Slope.
18. In the Subject to the Constraints box enter

$$V <= 0$$

$$V >= -20$$

This determines the range over which SOLVER will find the best fitting value of V. It can be seen from Fig. 2 that the value of V at $y = 0.5$ lies between 0 and -20. Constraints are used to impose limits over the range of values used to define the parameters. Although it is intuitive that the Slope is positive at $y = 0.5$, it is difficult to estimate the value so no constraints are applied to Slope.

19. Choose Solve to perform the fit. The programme will iteratively cycle through the fitting routine, changing the parameter values of V and Slope until the largest value of R^2 is calculated. These changes will be displayed on the spreadsheet template, as illustrated in Fig. 2B. The optimal values of V and Slope are -10.317 and 12.194, respectively, and the maximal value of R^2 is 0.997. The continuous thick line in Fig. 3B illustrates the best fit and it is clear that it is an improvement over the fit provided by the initial parameter values. Additionally the confidence intervals around the fit have been reduced.

2.5. Controlling advanced SOLVER features

The default SOLVER settings can be changed by opening the Solver Options Dialogue box (Fig. 4B). Each option has a default setting that is appropriate for most situations but that can be changed. The most relevant to the protocol described in this paper are described below.

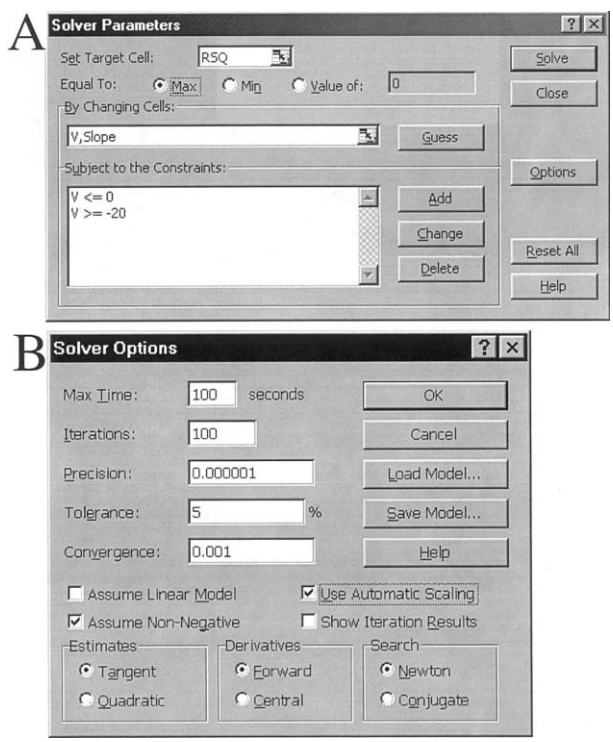

Fig. 4. The built-in SOLVER function. A: The SOLVER Dialogue box used as an interface between the SOLVER function and data on the spreadsheet. B: The fit can be fine-tuned using the Options Dialogue box.

Max time: Specifies the amount of time in seconds that SOLVER will be allowed to run before stopping. The default value is 100 s. Iterations: Specifies the number of iterations that SOLVER will carry out before stopping. The default value is 100. If SOLVER finds the optimal solution before either of these limits is reached it will present the results. Precision: Controls the precision of solutions by using the number entered to determine whether the value of a constraint meets a target or satisfies a lower or upper bound. The default value is 1×10^{-6}. The higher the precision, the more time taken to reach a solution. Tolerance: The percentage by which the target cell (RSQ in the example described here) of a solution satisfying the integer constraints can differ from the true optimal value and still be considered acceptable. This option applies only to problems with integer constraints. A higher tolerance tends to speed up the solution process. The default value is 5. Convergence: This value tells SOLVER when to stop the iterative process. When the relative change in the target cell value is less than the number in the Convergence box for the last five iterations, SOLVER stops. The smaller the convergence value, the more time SOLVER takes to reach a solution. The default value is 0.001. Assume Linear Model: This box should be checked only if the model to be solved in linear; otherwise, as in the case of the non-linear regression described here, leave the box unchecked. Use Automatic Scaling: Select to use automatic scaling when inputs and outputs have large differences in magnitude. For example, if values such as 1×10^{-14} are entered rounding off errors can be large. It is advised to keep this box checked for all SOLVER models. Assume Non-Negative: Causes SOLVER to assume a lower limit of 0 for all adjustable cells for which no constraints have been set. Show Iteration Results: Select to have SOLVER pause to show the results of each iteration. Estimates: Determines the approach used to obtain subsequent estimates of the basic variable values at the outset of each one-dimensional search. Tangent: Uses linear extrapolation from a tangent vector. Quadratic: The Quadratic choice extrapolates the minimum (or maximum) of a quadratic fitted to the function at its current point. The Tangent choice is slower but more accurate. Derivatives: Specifies the differencing used to estimate partial derivatives of the objective and constraint functions. Forward: The point from the previous iteration is used in conjunction with the current point. This reduces the recalculation time required for finite differencing, which can account for up to half of the total solution time. Central: Central differencing relies only on the current point and perturbs the decision variables in opposite directions from that point. Although this involves more recalculation time, it may result in a better choice of search direction when the derivatives are rapidly changing, and hence fewer total iterations. Search: Specifies the algorithm used at each iteration to determine the direction to search. Newton: The default choice Newton requires more memory but fewer iterations than does the Conjugate gradient method. Conjugate: Requires less memory than the Newton method but typically needs more iterations to reach a particular level of accuracy. Load Model: Loads a previously saved fitting routine. Save Model: Allows the user to save the current fitting routine for future use.

3. Conclusion

Non-linear regression is a powerful technique for standardizing data analysis. The advent of personal computers has rendered linear transformation of data obsolete, allowing non-linear regression to be carried out quickly and reliably by non-specialist users. While the method described in this paper requires that the user have a basic knowledge of spreadsheets, it is not required that the user has an intimate understanding of the mathematics behind the processes involved in curve fitting. This subject is beyond the knowledge of most biologists, involving calculus, matrices and statistics. What is important, however, is that the user understands enough about the data to be fit to use the correct type of analysis, and to judge goodness of fit from calculated estimates.

This paper does not address the issue of which functions are suitable to describe individual data,

but this topic is discussed in detail elsewhere where excellent guides to determining goodness of fit of a function using residual plots are described [2,5,6].

3.1. Assessment of goodness of fit

The R^2 value calculated in this paper is designed to give the user an estimate of goodness of fit of the function to the data, i.e. we assume that we are using an appropriate function to describe the data, but we want to know how accurately the function describes or fits the data. The R^2 value is called the coefficient of determination and its value represents the fraction of the overall variance of the 'dependent' variable that is explained by the 'independent' variable. It is calculated from the sum of the squares of the residuals and the sum of the squares of regression. The sum of the squares of the residuals captures the error between the estimate and the actual data and is analogous to the sum of the squares (within) in ANOVA (see the numerator of Eq. (8)). The sum of the squares of the residuals is used in linear regression to calculate the best fit (see earlier). The sum of the squares of regression calculates how far the predicted values differ from the overall mean, and is analogous to the sum of the squares (between) in ANOVA (see the denominator of Eq. (8)). In the example in this paper the R^2 value was 0.997 which means that 99.7% of the variation of the 'independent' variable can be explained by the variation of the 'dependent' variable.

After using SOLVER to calculate the converged values of the parameters one would like to know the reliability of those values. Some curve fitting programmes display the standard error of the parameters. However care should be taken in interpreting these values. As stated by Motulsky and Ransnas [6] "Non-linear regression programs generally print out estimates of the standard error of (the) parameters, but these values should not be taken too seriously. In non-linear functions, errors are neither additive nor symmetrical, and exact confidence limits cannot be calculated. The reported standard error values are based on linearizing assumptions and will always underestimate the true uncertainty of any non-linear equation…it is not appropriate to use the standard error values printed by a non-linear regression program in further formal statistical calculations." A method for calculating the asymptotic standard errors of the parameters has been devised, but it involves evaluating a Hessian matrix, a method that is "significantly more complex and requires significantly more computer time to evaluate. They also require a considerably more complex computer program" [2]. Thus the approach taken in this paper is to calculate the standard error of the data around the regression line, also known as the standard error of the residuals. This is calculated by dividing the sum of the squares of the residuals by the degrees of freedom to get the variance data about the regression line. Taking the square root of this value gives the standard error of the residuals. The standard error of the residuals can be used to calculate the confidence interval. The confidence interval is an indicator of the probability that the true value lies within the range specified by the probability formula. It is common to use 95% confidence interval, which means that there is a 95% probability that the true value lies within the interval. In order to calculate the confidence interval the Critical t-value must be calculated. This value depends on the confidence interval and the degrees of freedom. Fortunately Excel has a built-in function (tinv) which allows calculation of the Critical t-value, thus bypassing the need to look up tables of t values. The formula in cell H7 (1-confidence interval, degrees of freedom) calculates this value for our desired confidence interval and degrees of freedom. Once this value has been calculated the confidence interval is simply the best fit at all data points $\pm$ the Critical t-value*S.E. of residuals.

3.2. Advantages and limitations

While this protocol is regarded as robust and reliable a few points should be borne in mind.

First, the greater the number of parameters in the function the longer SOLVER will take. Additionally the more the user customizes the fitting protocol with additional constraints or increasing tolerance or precision, the longer SOLVER will take. Second, if initial parameter estimates are inappropriate, the iteration process can proceed in the wrong direction and a solution is not found. Thus it is important that sensible initial parameter estimates are input. Poor estimates may also lead to the wrong solution being found. This paper demonstrates an easily understood method for rapid fitting of data with non-linear functions.

References

[1] W.P. Bowen, J.C. Jerman, Nonlinear regression using spreadsheets, TiPS 16 (1995) 413–417.
[2] M.L. Johnson, Why, when, and how biochemists should use least squares, Anal Biochem. 206 (1992) 215–225.
[3] L.S. Lasdon, A.D. Waren, A. Jain, M. Ratner, Design and testing of a generalized reduced gradient code for nonlinear programming, ACM Trans. Mathematical Software 4 (1978) 34–50.
[4] S. Smith, L. Lasdon, Solving large sparse nonlinear programs using GRG, ORSA J. Comput. 4 (1992) 2–15.
[5] J. Dempster, Computer Analysis of Electrophysiological Signals, Academic Press, London, 1993.
[6] H.J. Motulsky, L.A. Ransnas, Fitting curves to data using nonlinear regression: a practical and nonmathematical review, FASEB J. 1 (1987) 365–374.