Nonlinear least-squares data fitting in Excel spreadsheets

Gerdi Kemmer & Sandro Keller

Leibniz Institute of Molecular Pharmacology FMP, Berlin, Germany. Correspondence should be addressed to S.K. (mail@sandrokeller.com).

Published online 28 January 2010; doi:10.1038/nprot.2009.182

We describe an intuitive and rapid procedure for analyzing experimental data by nonlinear least-squares fitting (NLSF) in the most widely used spreadsheet program. Experimental data in *x*/*y* form and data calculated from a regression equation are inputted and plotted in a Microsoft Excel worksheet, and the sum of squared residuals is computed and minimized using the Solver add-in to obtain the set of parameter values that best describes the experimental data. The confidence of best-fit values is then visualized and assessed in a generally applicable and easily comprehensible way. Every user familiar with the most basic functions of Excel will be able to implement this protocol, without previous experience in data fitting or programming and without additional costs for specialist software. The application of this tool is exemplified using the well-known Michaelis–Menten equation characterizing simple enzyme kinetics. Only slight modifications are required to adapt the protocol to virtually any other kind of dataset or regression equation. The entire protocol takes ~1 h.

INTRODUCTION

Modern measurement techniques allow researchers to gather ever more data in less time. In many cases, however, the primary or raw data have to be further analyzed, be it for the verification of a quantitative model (theory or hypothesis) thought to describe experimental data, quantitative comparison with other data, better visualization or simply data reduction. To this end, a wealth of information collected during a measurement or a series of measurements has to be reduced to a few characteristic parameters. This can be done by regression analysis, a statistical tool to find the set of parameter values that best describes the experimental data by assuming a certain relationship between two or more variables. Although many powerful and dedicated software packages have been developed for regression analysis, the most widely distributed regression tool is the Solver add-in bundled with Microsoft Excel.

We first describe the following important terms and concepts used in regression analysis: independent variable; dependent variable; dataset; regression equation; coefficient/parameter; best fit/solution; constraint; sum of squared residuals; simple, multiple, linear, and nonlinear regression; and starting/initial values. Then, we list some examples in which Solver was used to fit or simulate data. Specific advantages and disadvantages of Solver with respect to other data fitting programs as well as general limitations and pitfalls inherent in nonlinear regression analysis are also addressed. The core part of the protocol lays out a general fitting strategy and explains the application of Solver to a typical nonlinear regression problem encountered in biochemical research. Finally, we discuss a more difficult example to highlight some of the most common problems and pitfalls encountered in fitting experimental data.

Fundamentals of regression analysis

Quantitative experiments aim at characterizing a relationship between an *independent variable* (x), which is varied throughout a measurement, and a *dependent variable* (y_{obs}), which is observed/measured as a function of the former. The fitting method presented in this protocol requires that the independent variable can be measured with much greater precision than the dependent variable[1]. In other words, experimental errors (uncertainties) in the independent variable are small compared with errors in the dependent variable (see below). This is usually the case with experiments in which the value of the independent variable follows a predetermined trajectory and the experimental readout reports on the value of the dependent variable.

The primary output of a measurement is a set of conjugated independent and dependent variables, which is called *data* or *dataset*. In addition to an experimental dataset, regression analysis requires a *regression equation* (also termed *fitting function*). This is a mathematical relationship describing the dependence of the dependent variable on the independent variable using one or more parameters. These *parameters* (also called *adjustable parameters, fitting parameters or coefficients*) are the same for every data point, i (i.e., every combination of x_i and y_i). In the simplest example of a proportionality ($y = a \times x$), the only parameter, a, is the slope of a straight line through zero.

With these tools at hand and using a suitable computer program, one can employ regression analysis to find the combination of parameter values that best describes the experimental dataset. This combination of parameters is called the *best fit* or the *solution*. However, not every mathematically correct solution is physically possible or meaningful. For instance, temperature cannot fall below absolute zero. Mathematical relationships imposing such limits (e.g., $T(\mathrm{K}) > 0$, where T is the absolute temperature) are called *constraints* and can, or in some cases must, be provided in order for regression analysis to find a reasonable solution. Furthermore, constraints may help reduce the parameter value space that the regression routine has to sample, which is particularly useful for regression equations containing many parameters.

The decision as to which combination of parameters describes an experimental dataset best is most commonly done by *least-squares fitting* (LSF), i.e., by minimizing the *sum of squared residuals* (SSR). In this context, a *residual* is defined as the difference (δ) between an observed/measured data point (y_{obs}) and its calculated counterpart (y_{calc}). The sum runs over all data points to be considered for regression analysis. Provided that the experimental errors in the dependent variable follow a Gaussian (also known as normal) distribution with a mean of zero, the combination of parameter values having the lowest SSR has the highest probability of being correct. In a Gaussian distribution, small errors are more probable to occur than large ones. More precisely, for a very large number of independent replicate measurements, the frequency with which a certain error occurs follows a symmetrical bell-shaped function. A mean of zero implies that this distribution is centered at zero, such that errors are equally probable to be positive or negative. For many, but not all, experimental setups typically encountered in a biochemical or biophysical laboratory, errors (from pipetting, weighing, diluting, instrument noise, and so on) are or can be approximated to be distributed in a Gaussian way. For further discussion whether a particular dataset can be fitted reasonably using LSF, see reference 2.

Regression analysis comes in different forms: simple, multiple, linear and nonlinear. In *simple regression analysis*, there is only one independent variable, whereas *multiple regression analysis* handles several independent variables. In general, there can be as many dependent and independent variables as experimentally possible and reasonable (e.g., one can simultaneously record absorbance, conductivity and light scattering data when performing size exclusion chromatography). *Linear regression analysis* or *linear least-squares fitting* (LLSF) refers to regression equations that are linear in their parameters (this, of course, includes but is not limited to equations that are linear themselves). By contrast, *nonlinear regression analysis* or *nonlinear least-squares fitting* (NLSF) refers to equations that are nonlinear in their parameters. For example, $y = a \times \exp(x)$ is said to be linear in a because y is linearly proportional to a if the values of all other fitting parameters and of the independent variables (here, x) are held constant. Mathematically, this is manifested in the first derivative of y with respect to a being independent of a: $\mathrm{d}y/\mathrm{d}a = \exp(x)$. By contrast, $y = \exp(a \times x)$ is nonlinear in a because y is not linearly proportional to a if all other values are held constant. Here, differentiation of y with respect to a yields an expression dependent on a: $\mathrm{d}y/\mathrm{d}a = x \times \exp(a \times x)$. The solution to a linear regression problem is exact because it can be calculated by analytical means. By contrast, nonlinear regression cannot offer an analytical solution, but has to rely on iterative procedures to find the best fit. For further reading on regression analysis, see one of the pertinent books[3–6].

This protocol focuses on the more general case of nonlinear regression analysis. The underlying fitting algorithms are based on a trial-and-error approach: beginning with *starting values* (also called *initial values*) for the parameters, these are changed during every regression step, and the similarity between the measured and calculated values is analyzed and compared with the preceding iteration. The algorithm according to which the parameter values are altered and analyzed during each step depends on the fitting program used and some advanced settings. Excel Solver employs the generalized reduced gradient algorithm[7]; a discussion of the mathematical formalism is beyond the scope of this protocol but can be found elsewhere[8].

Published applications of Solver

Solver has been used in numerous and diverse applications, as illustrated by the following examples: modeling of detergent mixtures[9,10], the hypothalamic–pituitary–ovarian axis[11] or physiological functions[12]; estimation of protein-binding capacity[13] or magnetic relaxation times[14]; generation of complex pharmacokinetic models[15–17]; microbial population counting[18]; application of game theory to medical management[19]; analysis of percentile growth curves[20]; optimization of feed formulation in poultry science[21,22], operating-room allocation[23] or production of proteins[24], small molecules[25] or cheese[26]; calibration of ionization chambers[27]; analysis of electrophysiological measurements[28,29] or luminescence lifetime distributions[30]; deconvolution of chromatographic peaks[31–35] or thermoluminescence glow curves[36]; a number of applications in analytical chemistry[37–41]; and high-throughput fitting of dose–response curves[42].

We have employed Solver for deconvoluting absorbance and fluorescence spectra[43,44], fitting chemical shift perturbations in NMR titration experiments (Vargas *et al.*, unpublished data), simulating complex multicomponent/multiphase equilibria[45] or cooperative ligand binding to proteins[46], analyzing solubilization and reconstitution of lipid vesicles[47,48] (for an experimental protocol, see ref. 49), and monitoring lipid membrane partitioning and translocation with the aid of uptake and release experiments[44,50–55] (see ref. 56 for a protocol and an Excel spreadsheet implementing Solver). The latter application is a particularly impressive demonstration of the power of Solver: the interactions of many charged proteins, peptides and small molecules with lipid membranes can be described by a simple surface partition equilibrium modulated by electrostatic effects[57,58]. While the partition equilibrium yields the regression equation proper, membrane electrostatics have to be accounted for by Gouy–Chapman theory (or another suitable model). In brief, this means that during fitting of the actual regression equation to experimental data, a nonlinear constraint has to be fulfilled for each data point. This is a challenging task that is beyond the capabilities of many dedicated fitting programs but is easily handled by Solver (see ref. 50 for a detailed description of regression and constraint equations). Solver can also easily be called up from a macro executable in Excel (see **Box 1**).

Comparison with other fitting programs

Data analysis using NLSF can be accomplished with the aid of many different programs. Examples include Origin (OriginLab, Northampton, MA, USA), Prism (GraphPad Software, La Jolla, CA, USA) and PSI-Plot (Poly Software International, Pearl River, NY, USA), but there is a plethora of other commercial or free programs developed for this purpose. Of course, sophisticated mathematical software packages like Mathematica (Wolfram Research, Champaign, IL, USA) and MathLab (MathWorks, Natick, MA, USA) can carry out nonlinear regression, as well. Moreover, alternative approaches like genetic algorithms[59] may have to be implemented for more demanding problems, especially those containing numerous adjustable parameters. Most of these programs have a wide array of additional features like built-in statistical tests to assess the confidence of the best-fit parameter values or advanced fitting procedures like automated global fits. However, a major problem with specialized fitting programs is that they tend to entice non-expert users to adopt a black-box approach, so that they can hardly judge on the meaningfulness of the results returned by the program. In light of this, the more basic Solver may be a better option for most non-expert users wishing to fit experimental data by NLSF.

The procedure we describe here for Solver is simple, intuitive and fast, and can be implemented without previous knowledge of

BOX 1 | MACROS

Writing macros is beyond the scope of this protocol, but the following hints may aid users already familiar with this topic to incorporate Solver in Visual Basic for Applications (VBA) macros executable in Excel. Macros are particularly valuable for performing time-consuming and iterative tasks like confidence assessment (see Steps 11–18). In order to start the Solver add-in within a procedure, a reference must be added. To do this, click: TOOLS→REFERENCES→SOLVER in the Visual Basic Editor.

? TROUBLESHOOTING

Solver can then be called using the syntax
SolverOk SetCell: = 'B3', MaxMinVal: = 3, ValueOf: = '0', ByChange: = 'B1:B2'
SolverSolve True
Here, **SetCell: =** is equal to the **Set Target Cell** input in **Figure 4**, **MaxMinVal: = 3** is identified with checking **Value of:** (the third option in the **Equal To** selection; analogously, **1** stands for **Max** and **2** for **Min**), **ValueOf: =** specifies the target value, and **ByChange: =** tells the program which cells to vary during optimization. The parameters given above correspond to those displayed in **Figure 4** for our example. Defined names (see **Box 3** for details on naming cells) can be used instead of cell addresses (for instance, **ByChange: = 'v_max, K_m'** instead of the above). The term **SolverSolve True** is not necessary to run Solver but serves to replace manual confirmation (by clicking **OK**) upon completion of each fitting session. For further reading on advanced Excel solutions for scientific purposes, consult the excellent textbook of De Levie[1].

or experience in data fitting or programming. The vast majority of scientists are familiar with the basic tools offered by Excel, such as data manipulation or graphing facilities; in fact, a great deal of experimental data will eventually be transferred into this or a similar spreadsheet program for analysis or visualization. Excel is available in most laboratories and can be run on most personal computers or laptops, so no additional costs are incurred for dedicated software packages or workstations.

Limitations of regression analysis

General pitfalls of the fitting process itself have to be accounted for regardless of the fitting equation used. NLSF requires that all of the experimental errors can be attributed to the dependent variable and that the values of the independent variable be known precisely, that the data points be independent of one another, and that a sufficient number of data points be measured (see ref. 2 for a discussion of these and other important prerequisites for NLSF). Furthermore, a minimum in the SSR corresponds to a maximum in the likelihood of having found the correct parameter values only if the experimental errors in the dependent variable follow a Gaussian distribution with a mean of zero[2]. If this condition is not fulfilled, the minimal SSR does not represent the highest probability, and opinions diverge over the question if NLSF should still be used in such cases[2,3]. What is clear, however, is that systematic errors and improper data processing have to be avoided altogether. The latter includes nonlinear operations like taking a logarithm or smoothing, which diminishes the information content of the data. By contrast, linear operations like addition or subtraction (e.g., of blank values), as well as multiplication or division by a constant factor (e.g., normalization), are permitted. Moreover, one should keep in mind that NLSF is a trial-and-error approach and that there can never be an absolute certainty of having found the global SSR minimum unless a systematic sampling of all possible combinations of parameter values is carried out. Another matter of concern is weighting (see **Box 2**).

These issues aside, the results of a fitting session have to be interpreted cautiously and in the light of the specific problem at hand. Not every parameter mathematically extractable through regression analysis makes sense physically (see ANTICIPATED RESULTS for an example), and not every regression equation allowing for a

BOX 2 | WEIGHTING

The use of an unweighted SSR implies that the experimental errors of all data points included in the fit follow the same Gaussian distribution having the same mean (μ = 0) and the same standard deviation (σ). There are, however, scenarios where σ varies considerably over the range of recorded data. This should be accounted for by giving different data points a different weight, i.e., a different impact on the SSR.

In its most general form, a weighted SSR is given by: $\text{SSR} = \sum_i w_i (y_{i,\text{obs}} - y_{i,\text{calc}})^2$, where w_i is the weighting factor for the *i*th data point. Thus, data points with higher weighting factors contribute more to the SSR than those given less weight. The special case of w_i = 1 for all *i* corresponds to an unweighted SSR.

The question then remains as to a meaningful and justified choice of weighting factors. *Weighting by observed variability* is a theoretically simple approach relying on the standard deviation of the *i*th data point, σ_i: $\text{SSR} = \sum_i ((y_{i,\text{obs}} - y_{i,\text{calc}})/\sigma_i)^2$. This weighted SSR is also known as χ^2 value. From a practical viewpoint, however, this weighting method can be of limited utility because it might require that the standard deviation for each data point be determined from a fairly large number of replicate experiments (usually several dozen). Another important weighting scheme is *relative weighting*: $\text{SSR} = \sum_i ((y_{i,\text{obs}} - y_{i,\text{calc}})/y_{i,\text{obs}})^2$. This approach is practically straightforward and is applicable whenever the error scales with the value of the measured variable. Discussions and comparisons of weighting methods can be found in the literature[5,60].

Once appropriate weighing factors have been determined, they can easily be included in Step 6 of the standard protocol by putting them into column H, changing the formula in column G2 to **= H2*(E2–F2)^2**, and copying or dragging this formula down into the following cells in column G.

good fit is an appropriate description of the underlying relationship between independent and dependent variables. These precautions, however, pertain not to the fitting process as such but to the choice of regression equation.

General fitting strategy

Data fitting with the aid of Solver can be divided into the following stages: (i) an Excel worksheet is set up. Experimental data are pasted and plotted, and simulated data are calculated using a regression equation and plotted, too. If applicable and necessary, data processing like normalization and baseline subtraction should be completed first. The squared residuals between observed and calculated data are computed and summed up. Before moving on, the adjustable parameters should be varied to get a feel of the influence of each parameter on the calculated curve and to determine reasonable starting values. (ii) The Solver add-in is prepared. Cells containing the values to be changed (the fitting parameters) and the value to be minimized (the SSR) are specified. Solver options may be adapted and constraints can be added if necessary. (iii) The fitting procedure is carried out by running Solver. This is repeated several times using different starting values for the adjustable parameters. Although this approach cannot guarantee finding the global SSR minimum, it does decrease the probability of unwittingly getting stuck in a local minimum. (iv) The confidence of the best-fit parameter values is assessed by repeating the fitting procedure while fixing the parameter to be scrutinized at a value slightly different from the optimal one.

After determining the best-fit parameter values, two important questions have to be answered: how good is the agreement between experimental data and fit, and how much confidence can be put in the best-fit parameter values returned by nonlinear regression? These two issues are sometimes confused, but it is crucial to appreciate the fundamental difference between them. The question of goodness of fit refers to the extent to which a dataset calculated assuming a certain fitting equation can approach the experimental dataset. The SSR and related (i.e., weighted or normalized) quantities, such as the χ^2 value (see **Box 2**) or the root-mean-square deviation (RMSD), are measures of the goodness of fit. For a given dataset and fitting equation, the goodness of fit generally increases with decreasing number of data points or increasing number of fitting parameters. Most importantly, the goodness of fit contains absolutely no information about the confidence of the fitted parameter values, which generally increases with increasing number of data points or decreasing number of fitting parameters.

Thus, a different quantity is needed to describe confidence. To this end, virtually all commercial NLSF programs compute some kind of standard error or standard deviation which is displayed along with the corresponding best-fit value. Automatic generation of such confidence intervals might appear convenient but can be highly misleading. The underlying calculations are based on a series of linear approximations that are never fulfilled for NLSF. This invariably results in an underestimation of the real uncertainties[60,61], which becomes particularly problematic for small datasets and strongly correlated parameters[5]. A robust solution to this problem that can easily be implemented in a spreadsheet program consists in perturbing one of the fitting parameters from its best-fit value and recording how the SSR is affected on fitting the remaining parameters[2,5,62,63]. Parameter values for which the SSR exceeds a certain threshold can then be used as measures of confidence for that parameter. Using this approach may be less convenient than relying on statistical parameters reported by other fitting programs. However, it provides a more realistic picture and a much deeper understanding of the confidence with which the desired information can be extracted from the experimental data at hand (see ANTICIPATED RESULTS for an example in which commercial fitting programs suggest a misleadingly narrow confidence interval for an extremely poorly defined parameter).

Example of nonlinear regression: enzyme kinetics

A prominent example of a regression equation that is nonlinear in its parameters is the Michaelis–Menten equation[64] describing enzyme kinetics. The Michaelis–Menten equation gives the initial velocity of an enzymatic reaction, v, in dependence on the concentration of substrate, [S]: $v = v_{\max}[\mathrm{S}]/(K_{\mathrm{m}} + [\mathrm{S}])$. Here $v_{\max}$ is the maximal velocity (for $[\mathrm{S}] \to \infty$), and K_{m} is the Michaelis–Menten constant (i.e., the substrate concentration at which the initial velocity is half the maximal velocity). In this example, [S] is the independent variable (x in generic terminology), v is the dependent variable (y), and $v_{\max}$ and K_{m} are the fitting/adjustable parameters. Reasonable constraints for this case would be $v_{\max} \geq 0$ mM s^{-1} and $K_{\mathrm{m}} > 0$ mM to avoid division by zero for the first data point (in the absence of substrate, [S] = 0 mM).

In times when computers were not yet omnipresent, linearization approaches like the Lineweaver–Burk plot[65] were frequently employed to circumvent nonlinear regression analysis. Although such plots may be of great didactic value, their principal drawback is that processing data in any nonlinear way (i.e., subjecting them to operations other than, e.g., subtraction of another dataset or multiplication by a constant factor) not only transforms the

TABLE 1 | Simulated data used for the example discussed in the protocol.

[S] (mM)	v(mM s^{-1})
0	−2
2.5	153
5	231
10	342
15	396
20	438
30	467
40	505
50	523
60	523
70	539
80	548
90	555
100	554

Data points were simulated using the Michaelis–Menten equation and $v_{\max}$ and K_{m} values determined for the conversion of carbon dioxide to carbonic acid catalyzed by carbonic anhydrase at an enzyme concentration of 1 μM (taken from ref. 66). A Gaussian random-error term was added to simulate experimental errors. [S], substrate concentration (independent variable, x); and v, initial reaction velocity (dependent variable, y_{obs}).

measured values but also distorts the associated experimental errors (see ref. 62 for an example). In brief, linearization of data to render them amenable to linear regression can have adverse effects on the determination of the best-fit parameter values and, therefore, should be avoided. Nowadays, hard- and software necessary for nonlinear regression has become available to most scientists and has thus obviated the need for linearization.

This protocol provides a step-by-step guide on how to extract v_{max} and K_m from the dataset in **Table 1** through nonlinear regression and how to assess the confidence of the best-fit values using Microsoft Excel Solver. The dataset was simulated on the basis of values determined for the conversion of carbon dioxide to carbonic acid catalyzed by carbonic anhydrase[66]; a Gaussian random-error term was added to simulate experimental noise.

MATERIALS

EQUIPMENT

- Computer equipped with Microsoft Excel 3.0 or newer. Excel Solver was developed by Frontline Systems and has been included in every distribution of Microsoft Excel since 1990. Solver is not included in the newest version of Excel for Mac (Excel 2008) but can be downloaded free of charge from http://www.solver.com/mac/dwnmacsolver.htm. This protocol was prepared using Office 2007; click paths may vary slightly depending on the Excel distribution used.
- Experimental data in *x*/*y* form. In the following, we exemplarily use the dataset in **Table 1**, which was simulated using the Michaelis–Menten equation including a Gaussian random-error term.
- Regression equation describing the data; here, the Michaelis–Menten equation given in the last section of the INTRODUCTION.

PROCEDURE

Activating Solver ● TIMING 1 min

1| Open an Excel workbook.

2| Click OFFICE BUTTON → EXCEL OPTIONS → ADD-INS; in the MANAGE drop-down menu, choose EXCEL ADD-INS and hit the **Go** button (Excel 2003: TOOLS → ADD-INS). In the ADD-INS window, check the **Solver** checkbox (see **Fig. 1**) and click **OK**. When asked to confirm, choose **Yes**.
? TROUBLESHOOTING

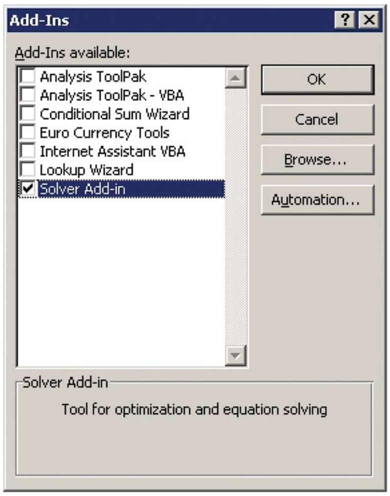
Setting up worksheet and plotting data ● TIMING 10 min

3| Type $\boldsymbol{v_{max}}$ (the first adjustable parameter of the Michaelis–Menten model) into cell A1 and define the name of cell B1 as **v_max** (see **Box 3** for instructions on how to name cells). Type $\boldsymbol{K_m}$ (the second adjustable parameter) into cell A2 and define the name of cell B2 as **K_m**. Set v_{max} to 400 mM s^{-1} and K_m to 2.0 mM by typing **400** and **2** into cells B1 and B2, respectively. See **Figure 2** for clarification.

4| Use the first row of columns D, E, F and G to denote the corresponding columns as ***x* (mM)**, $\boldsymbol{y_{obs}}$ **(mM s^{-1})**, $\boldsymbol{y_{calc}}$ **(mM s^{-1})** and $\boldsymbol{\delta^2}$ **(mM^2 s^{-2})**, respectively. Plug the data listed in **Table 1** into columns D and E (*x* in column D and y_{obs} in column E). Graph the data as scatter plot and label the axes. In our example, *x* is the substrate concentration, [S], and y_{obs} is the initial reaction velocity, *v*.

5| Type the regression equation, **=v_max*D2/(K_m+D2)** into cell F2. Copy or drag this formula down into the following rows of the column, which will be filled with the corresponding y_{calc} values. Add the calculated data to the graph as line plot using a different color for clearer discrimination between observed and calculated data. **Figure 2** shows the resulting Excel worksheet.

6| Type **=(E2−F2)^2** into cell G2 and copy or drag this equation down into the rest of the column. This is the squared residual (δ^2) between the observed data and data calculated using the regression equation. Type **SSR** into cell A3 and **=SUM(G2:G15)** (or the corresponding function name in your Excel language) into cell B3, which now displays the sum of the values listed in column G. The spreadsheet in FORMULA AUDITING MODE is shown in **Figure 3** (FORMULA AUDITING MODE can be switched on/off in FORMULAS → SHOW FORMULAS (Excel 2003: TOOLS → FORMULA AUDITING → FORMULA AUDITING MODE)).
▲ CRITICAL STEP As an alternative to Step 6, the SSR can be calculated directly by typing **=SUMXMY2(E2:E15;F2:F15)** into cell B3. It should, however, be realized that this function has rather different names in non-English versions of Excel.



Add-Ins

Add-Ins available:

□ Analysis ToolPak
□ Analysis ToolPak - VBA
□ Conditional Sum Wizard
□ Euro Currency Tools
□ Internet Assistant VBA
□ Lookup Wizard
☑ Solver Add-in

Solver Add-in

Tool for optimization and equation solving

Figure 1 | ADD-INS window. Make sure Solver is checked. To call this window: OFFICE BUTTON → EXCEL OPTIONS → ADD-INS; in the MANAGE menu, choose EXCEL ADD-INS and hit the **Go** button (Excel 2003: TOOLS → ADD-INS).

BOX 3 | NAMING CELLS

Naming cells is a particularly useful feature in carrying out spreadsheet calculations and fitting or simulating data. In contrast to referencing between cells, naming cells lacks the need for protection of absolute references (i.e., references that should not change on dragging or copying). Naming saves time and contributes to clarity and transparency, especially when typing lengthy formulas containing many variables and coefficients. Naming can be done either by using the menu command FORMULAS→DEFINE NAME (Excel 2003: INSERT→NAMES→CREATE) or by highlighting a cell or cell range and typing the desired name into the LABEL FIELD, located in the TOOLBAR directly above cell A1.

Names can be assigned to single cells or ranges of cells. The name of a single cell is displayed in the LABEL FIELD whenever the cell is activated (e.g., the LABEL FIELD in **Fig. 2** displays **v_max** as cell B1 is activated). The name of a cell range appears in the LABEL FIELD only when the entire cell range is activated. For ranges spanning several cells in the same row (or column), the calculator will refer to the cell in the same column (or row) as the formula. Names are dragged and dropped with their respective cells. Names can be edited or deleted using FORMULAS→NAME MANAGER (Excel 2003: INSERT→NAMES).

No two cells can have the same name, thus naming a cell **D3** will not work because this name is already given to cell D3 in the third row of the fourth column. When trying to define a name that is already used within the same workbook, the cell to be named will remain unnamed, and the previously named cell will be highlighted instead. For further information, consult the Excel manual or one of the numerous pertinent books.

Setting up solver ● TIMING 10 min

7| Click DATA → SOLVER (Excel 2003: TOOLS → SOLVER). This opens the SOLVER PARAMETERS window, prompting for entries for **Set Target Cell**, **Equal To**, **By Changing Cells** and **Subject to the Constraints**. **Figure 4** shows the entries appropriate for our example. **Set Target Cell** determines which value is to be optimized. When using the least-squares procedure, this is the SSR in cell B3, which has to be minimized. This is accomplished by opting for **Value of: 0** in the **Equal To** selection (see below). **By Changing Cells** defines the adjustable parameters, which in our case are v_{max} and K_m contained in cells B1 and B2, respectively. **Subject to the Constraints** offers the relational operators **>=**, **<=**, **=**, **int** and **bin** to impose constraints on the adjustable parameters or any other variables in the spreadsheet. **int** allows only integer values for the specified variable, and **bin** allows only binary values. Constraints can be added, changed or deleted by clicking the corresponding buttons. A useful constraint in the present example is $\boldsymbol{K_m}$ **>= 0.000001** to prevent division by zero for the first data point recorded in the absence of substrate ([S] = 0 mM).

▲ CRITICAL STEP Minimization of the target-cell value can, in principle, be carried out by checking the **Min** checkbox in the **Equal To** selection. However, we have noticed in numerous cases that Solver is more likely to get stuck in a local minimum when using this option. Upon checking **Value of: 0**, the algorithm tries harder to reduce the SSR to zero. As this is not perfectly feasible with experimental data, Solver will open a window stating that it was unable to find a solution. This message can be ignored.

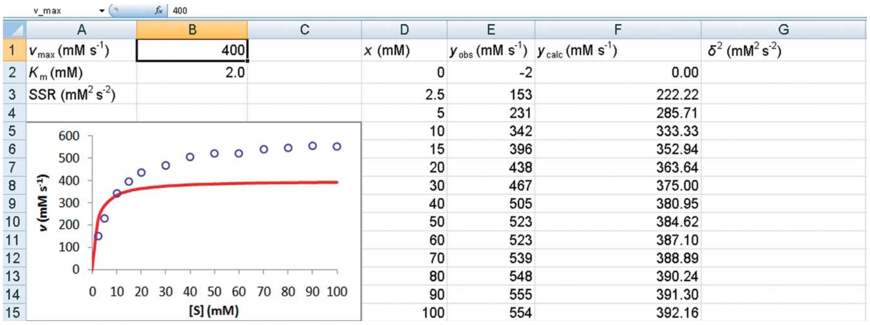
▲ CRITICAL STEP More sophisticated Solver settings are available in the SOLVER OPTIONS window, which is accessible by clicking the OPTIONS button in the SOLVER PARAMETERS window. **Figure 5** shows the default values, which work well in most cases. The meanings of these parameters are explained in **Box 4**.

? TROUBLESHOOTING

Fitting ● TIMING 10 min

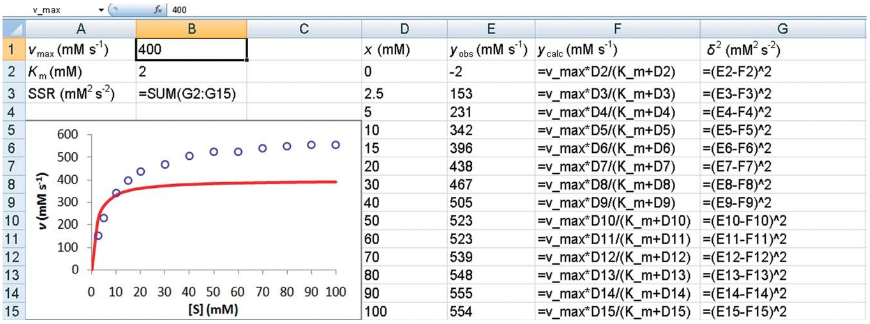
8| In the SOLVER PARAMETERS window (see **Fig. 4**), click the **Solve** button to initialize Solver. At the end of the fitting procedure, the SOLVER RESULTS window opens, reporting that no feasible solution could be found. As explained in Step 7, this is because of our selection in the SOLVER PARAMETERS window and can be ignored. One can also choose to view additional information on the fitting procedure, which in most cases is little enlightening, and therefore can be skipped. Hit the **OK** button in the SOLVER RESULTS window.

? TROUBLESHOOTING



	A	B	C	D	E	F	G
1	v_{max} (mM s^{-1})	400		x (mM)	y_{obs} (mM s^{-1})	y_{calc} (mM s^{-1})	δ^2 (mM^2 s^{-2})
2	K_m (mM)	2.0		0	-2	0.00	
3	SSR (mM^2 s^{-2})			2.5	153	222.22	
4				5	231	285.71	
5				10	342	333.33	
6				15	396	352.94	
7				20	438	363.64	
8				30	467	375.00	
9				40	505	380.95	
10				50	523	384.62	
11				60	523	387.10	
12				70	539	388.89	
13				80	548	390.24	
14				90	555	391.30	
15				100	554	392.16	

Figure 2 | Worksheet after the completion of Step 5. Highlighted cell B2 is named **v_max**, as can be seen in the LABEL FIELD in the upper left. Columns D and E are filled with, respectively, x and y_{obs} data from **Table 1**, and y_{obs} is plotted against x (blue scatter plot). Column F contains the values calculated using the parameters in cells B1 and B2 and the Michaelis–Menten equation, y_{calc}. Calculated values are plotted against x as solid red line.



v_max | 400

	A	B	C	D	E	F	G
1	v_{max} (mM s^{-1})	400		x (mM)	y_{obs} (mM s^{-1})	y_{calc} (mM s^{-1})	δ^2 (mM^2 s^{-2})
2	K_m (mM)	2		0	-2	=v_max*D2/(K_m+D2)	=(E2-F2)^2
3	SSR (mM^2 s^{-2})	=SUM(G2:G15)		2.5	153	=v_max*D3/(K_m+D3)	=(E3-F3)^2
4				5	231	=v_max*D4/(K_m+D4)	=(E4-F4)^2
5				10	342	=v_max*D5/(K_m+D5)	=(E5-F5)^2
6				15	396	=v_max*D6/(K_m+D6)	=(E6-F6)^2
7				20	438	=v_max*D7/(K_m+D7)	=(E7-F7)^2
8				30	467	=v_max*D8/(K_m+D8)	=(E8-F8)^2
9				40	505	=v_max*D9/(K_m+D9)	=(E9-F9)^2
10				50	523	=v_max*D10/(K_m+D10)	=(E10-F10)^2
11				60	523	=v_max*D11/(K_m+D11)	=(E11-F11)^2
12				70	539	=v_max*D12/(K_m+D12)	=(E12-F12)^2
13				80	548	=v_max*D13/(K_m+D13)	=(E13-F13)^2
14				90	555	=v_max*D14/(K_m+D14)	=(E14-F14)^2
15				100	554	=v_max*D15/(K_m+D15)	=(E15-F15)^2

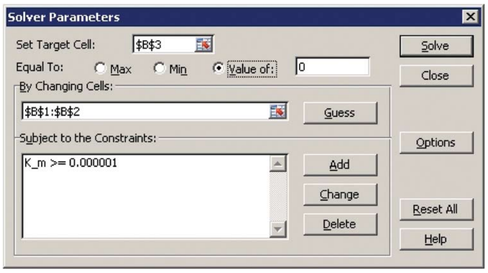
Figure 3 | Worksheet in FORMULA AUDITING MODE after the completion of Step 6. Cell B3 contains the sum of squared residuals (SSR), column F the values calculated using the Michaelis–Menten equation (y_{calc}) and column G the squared residuals (δ^2). y_{obs} and y_{calc} are plotted against x (blue scatter plot and solid red line, respectively). FORMULA AUDITING MODE can be switched on/off by clicking FORMULAS → SHOW FORMULAS (Excel 2003: TOOLS → FORMULA AUDITING → FORMULA AUDITING MODE).

9| Repeat the fitting procedure by clicking DATA → SOLVER (Excel 2003: TOOLS → SOLVER) and hitting the **Solve** button until the SSR in cell B3 no longer decreases.
? TROUBLESHOOTING

10| Inspect the x/y plot containing the experimental data and the data calculated using the best-fit parameter values returned by Solver: v_{max} = 598 mM s^{-1} and K_m = 7.6 mM in cells B1 and B2, respectively. Experimental and fitted data should resemble one another reasonably well, as illustrated in **Figure 6**. This is reflected in a small SSR value of ~298 mM^2 s^{-2}.
▲ CRITICAL STEP Once the best fit for a given set of initial parameter values has been found, it is highly advisable to repeat the fitting procedure using different starting values for the adjustable parameters to reduce the risk of getting stuck in a local rather than the global SSR minimum.
▲ CRITICAL STEP Visual inspection of the goodness of fit is often facilitated by plotting the residuals (not the squared residuals) versus the independent parameter. This helps in judging the quality of the fit and the applicability of the chosen regression equation (see ref. 67 for an example of analysis of residual plots).
▲ CRITICAL STEP The steps outlined thus far yield the v_{max} and K_m values that best fit the experimental data in **Table 1** in terms of the Michaelis–Menten model. Yet, this procedure affords no information on the confidence to put in the obtained values. In contrast to linear regression, simple statistical measures like the standard error or the standard deviation are not readily applicable to nonlinear regression analysis. In the following, we describe a straightforward and general way of assessing confidence by variation of the SSR near a minimum[2,5,62,63].
? TROUBLESHOOTING

Confidence assessment ● TIMING 30 min (15 min per parameter)

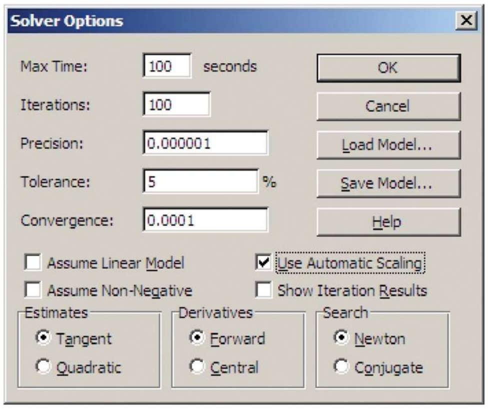
11| The basic idea of variation of the SSR near a minimum is to fix one of the adjustable parameters at various values close to but different from the optimal solution and to monitor the impact on the SSR on optimizing the other parameters. To this end, fix the value of K_m at 7.4 mM by typing **7.4** into cell B2 while leaving the value of v_{max} in cell B1 unchanged. Click DATA → SOLVER (Excel 2003: TOOLS → SOLVER) and select only cell B1 in the **By Changing Cells** option in the SOLVER PARAMETERS window. Then, hit the **Solve** button and repeat this one-parameter fit until the SSR in cell B3 no longer decreases. In this example, the SSR will amount to 330 mM^2 s^{-2}. Copy this value as well as the fixed K_m and fitted v_{max} values into another worksheet using PASTE SPECIAL: VALUES.



Solver Parameters

Set Target Cell: $B$3
Equal To: ○ Max ○ Min ● Value of: 0
By Changing Cells:
$B$1:$B$2
Subject to the Constraints:
K_m >= 0.000001

Figure 4 | SOLVER PARAMETERS window with the appropriate values for **Set Target Cell**, **Equal To**, **By Changing Cells** and **Subject to the Constraints**. An equally appropriate entry for **By Changing Cells** would be **K_m, v_max**. To call this window: DATA → SOLVER (Excel 2003: TOOLS → SOLVER).



Solver Options

Max Time: 100 seconds
Iterations: 100
Precision: 0.000001
Tolerance: 5 %
Convergence: 0.0001
☐ Assume Linear Model
☑ Use Automatic Scaling
☐ Assume Non-Negative
☐ Show Iteration Results
Estimates: ● Tangent ○ Quadratic
Derivatives: ● Forward ○ Central
Search: ● Newton ○ Conjugate

Figure 5 | SOLVER OPTIONS window with default selection of values appropriate for most fitting problems. See **Box 4** for further information on the settings.

BOX 4 | SOLVER OPTIONS

The default values shown in the `SOLVER OPTIONS` window in **Figure 5** are a good starting point for most fitting tasks. **Max Time** and **Iterations** define the time and number of iterations, respectively, after which Solver interrupts the fitting procedure and asks whether it should continue. Irrespective of these two parameters, Solver can be halted any time during the fitting procedure by pressing the `ESC` key. The value given in the **Precision** box determines how strictly constraints have to be fulfilled, whereas **Tolerance** specifies how strictly the integer criterion will be applied (if applicable at all). **Convergence** determines when optimization is considered to be achieved: this is the case as soon as five consecutive iterations do not yield a relative change in the target-cell value greater than the value specified in this box. If **Assume Linear Model** is checked, Solver uses linear regression and cannot solve nonlinear problems. **Assume Non-Negative** can be checked whenever none of the adjustable parameters is allowed to become negative. If **Use Automatic Scaling** is enabled, Solver changes the parameter values in relation to the magnitude of the initial values, which is recommended for all applications. Checking **Show Iteration Results** causes Solver to pause after every iteration and display the current values. The **Estimates**, **Derivatives** and **Search** options determine the details of the regression process. For difficult problems, it may be advantageous to choose **Central** in the **Derivatives** option. See the literature[7,28] for more information on these advanced settings.

12| Repeat Step 11 by decreasing the fixed K_m value in 0.2-mM increments until the SSR is more than four times greater than the best-fit SSR, i.e., >1200 $mM^2\ s^{-2}$. This should be the case for K_m = 6.4 mM, for which the SSR amounts to 1547 $mM^2\ s^{-2}$.

13| Starting from a K_m value of 7.8 mM, repeat Step 11 by increasing the fixed K_m value in 0.2-mM increments until the SSR is more than four times greater than the best-fit SSR, i.e., >1200 $mM^2\ s^{-2}$. This should be the case for K_m = 8.8 mM, for which the SSR amounts to 1225 $mM^2\ s^{-2}$.

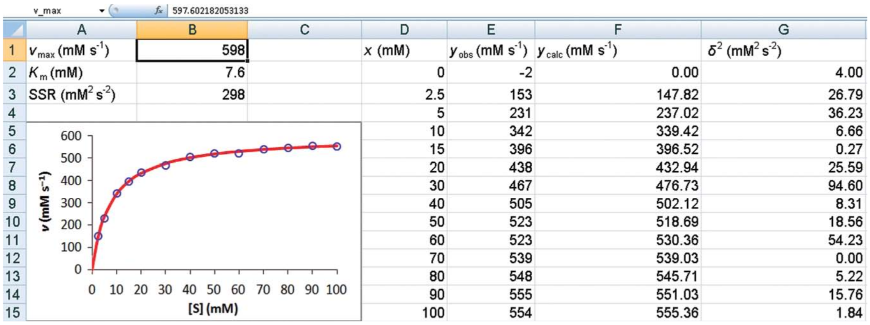
14| Plot the SSR values calculated in Steps 9 and 11–13 against the corresponding K_m values. The resulting diagram should show a well-defined minimum at K_m = 7.6 mM, as depicted in **Figure 7a**. The steepness of the SSR curve on both sides of the minimum is a measure of the precision with which the best-fit value can be determined from the given experimental data. It is obvious from this plot that the SSR rises sharply (i.e., the goodness of fit decreases dramatically) upon small deviations from the optimal K_m, indicating that K_m can be determined from the given dataset with high confidence.

▲ CRITICAL STEP Plots of the SSR against one or several adjustable parameters are sometimes rather shallow or extremely asymmetric (see ANTICIPATED RESULTS for an example). In such cases, the following Steps (15, 16 and 18) employed to derive confidence intervals are not applicable (see **Box 5**). This emphasizes the importance of visual inspection and judicious interpretation of SSR plots.

▲ CRITICAL STEP Instead of plotting the SSR, the value of the second adjustable parameter (v_{max}, which is varied in Steps 11–13) can be plotted against that of the 'frozen' parameter. This gives an idea about the correlation (interdependence) of the two parameters, i.e., if one parameter changes vigorously upon alteration of the other, then the two parameters are highly correlated (interdependent). This can be visualized in a contour plot or a three dimensional graph (see refs. 2,5,62,63 for examples).

? TROUBLESHOOTING

15| In cases where the SSR minimum is defined as clearly as in the present example, true confidence intervals can be approximated by the following procedure: to calculate a 95% confidence interval for K_m, type **=B3*(1 + 2/12*FINV(1-95/100; 2; 12))** into an empty cell in the Excel workbook. This should return a so-called threshold value of 491 $mM^2\ s^{-2}$. In the SSR plot, draw a horizontal line at SSR = 491 $mM^2\ s^{-2}$ and determine the two K_m values at which this straight



v_max

597.602182053133

	A	B	C	D	E	F	G
1	v_{max} (mM s^{-1})	598		x (mM)	y_{obs} (mM s^{-1})	y_{calc} (mM s^{-1})	δ^2 (mM^2 s^{-2})
2	K_m (mM)	7.6		0	-2	0.00	4.00
3	SSR (mM^2 s^{-2})	298		2.5	153	147.82	26.79
4				5	231	237.02	36.23
5				10	342	339.42	6.66
6				15	396	396.52	0.27
7				20	438	432.94	25.59
8				30	467	476.73	94.60
9				40	505	502.12	8.31
10				50	523	518.69	18.56
11				60	523	530.36	54.23
12				70	539	539.03	0.00
13				80	548	545.71	5.22
14				90	555	551.03	15.76
15				100	554	555.36	1.84

Figure 6 | Worksheet after the completion of Step 10. Cells `B1` and `B2` contain the best-fit values for v_{max} and K_m, respectively. The solid red line represents the best fit (y_{calc}) to the measured data (y_{obs}; blue scatter).

line intersects the SSR curve (light gray lines in **Fig. 7a**). The lower and upper values amount to 7.1 and 8.1 mM, respectively. Thus, the 95% confidence interval of K_m can be approximated as 7.1–8.1 mM. See **Box 5** for further details.

16| Repeat Step 15 for a 99% confidence interval by using the formula **=B3*(1 + 2/12*FINV(1-99/100; 2; 12))**. The threshold value now is 642 mM^2 s^{-2}, which intersects the SSR curve at K_m values of 7.0 and 8.3 mM (dark gray lines in **Fig. 7a**). Thus, the 99% confidence interval of K_m can be approximated as 7.0–8.3 mM. See also **Box 5**.

17| Repeat Steps 11–14 with the second adjustable parameter to obtain a plot of the SSR versus v_{max}. In order to span the same range of SSR values as above (Steps 12 and 13), v_{max} should be fixed at values ranging from 580 to 620 mM s^{-1} using increments of 5 mM s^{-1}. The resulting diagram is presented in **Figure 7b**. Again, the steep slopes on both sides of the minimum indicate that v_{max} is determined with high confidence.
? TROUBLESHOOTING

18| Repeat Steps 15 and 16 with the second adjustable parameter to derive 95% and 99% confidence intervals of v_{max}. To this end, use the threshold SSR values calculated above (491 and 642 mM^2 s^{-2}, for 95% and 99% confidence intervals, respectively) and determine the v_{max} values at which the corresponding horizontal lines intersect the SSR curve (gray lines in **Fig. 7b**). The confidence intervals thus obtained should be 590–606 mM s^{-1} at 95% and 587–609 mM s^{-1} at 99% confidence. See also **Box 5**.

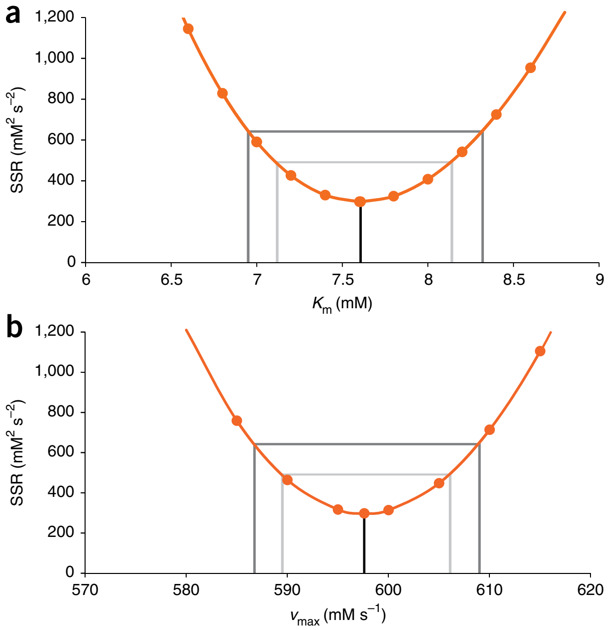


Figure 7 | Confidence assessment of best-fit parameter values. Plots of sum of squared residuals (SSR) against (**a**) K_m and (**b**) v_{max}. Light and dark gray lines mark parameter values at which the SSR amounts to, respectively, 491 and 642 mM^2 s^{-2}. These parameter ranges correspond to the approximate 95% and 99% confidence intervals, respectively. Vertical black lines indicate best-fit values.

● **TIMING**

Steps 1–2, Activating Solver: 1 min (provided Solver is installed)
Steps 3–6, Setting up worksheet and plotting data: 10 min
Step 7, Setting up Solver: 10 min
Steps 8–10, Fitting: 10 min (may vary depending on the performance of the computer being used)
Steps 11–18, Confidence assessment: 30 min (15 min per parameter)

BOX 5 | CALCULATION OF CONFIDENCE INTERVALS USING FISHER'S *F* DISTRIBUTION

The connection between an SSR plot (see **Fig. 7**) and lower and upper confidence intervals at a desired confidence level (*P*, expressed in percent) is established by a threshold SSR value (SSR_{th}). Under certain conditions (as explained below), the latter can be calculated from the best-fit SSR value (SSR_{bf}) on the basis of Fisher's *F* distribution[2,5,62,63]: $\mathrm{SSR_{th}} = \mathrm{SSR_{bf}} \cdot \left(1 + \frac{M}{N-M} \cdot F(1-P/100;M;N-M)\right)$. *F* is the so-called upper (1–*P*/100) quantile of Fisher's *F* distribution with *M* being the number of adjustable parameters and *N* the number of data points included in the fit. The difference *N*–*M* is also referred to as the number of degrees of freedom. *F* can easily be calculated in Excel using the formula **=FINV(1–P/100; M; N–M)**, as shown in Steps 15 and 16 for $M = 2$ adjustable parameters (K_m and v_{max}) and $N = 14$ independent data points at confidence levels of 95 and 99%, respectively.

Importantly, application of Fisher's *F* distribution is strictly valid only for linear fitting equations[2,63]. However, if an SSR plot reveals a clear minimum with steep slopes on both sides (as in **Fig. 7**), even a nonlinear fitting equation can be assumed to be approximately linear for small deviations from the best-fit value. Then, confidence intervals at a certain confidence level *P* can be approximated according to the above procedure (see Steps 11–18). By contrast, if the minimum in SSR is not that well defined (as in **Fig. 9**), this approximation no longer holds. Nevertheless, lower and upper parameter values can be derived from an arbitrarily chosen threshold SSR, but there are no simple means of ascribing a confidence level to parameter ranges thus obtained (see ANTICIPATED RESULTS).

? TROUBLESHOOTING

Troubleshooting advice can be found in **Table 2**. Further information on specific Solver messages can be found online at http://www.solver.com/suppstdmessages.htm.

TABLE 2 | Troubleshooting table.

Step	Problem	Possible reason	Solution
2	Solver is not listed in the ADD-INS window	Solver is not installed	Install the Solver add-in from the Microsoft Office CD/DVD
7	Solver cannot be found at the specified location	Solver is not activated as an add-in	Activate the Solver add-in as described in Step 2
		A graph window is activated	Activate any spreadsheet cell. Solver can be run only from a spreadsheet
8,9	Error message: 'Solver cannot find a solution to the problem'	In the SOLVER PARAMETERS window, **Value of: 0** is checked in the **Equal To** selection	No problem. Provided that the fitted curve resembles the experimental data, ignore the error message. See Step 7 for details
	Error message: 'The problem is too large for Solver to handle'	The Solver version shipped with Microsoft Excel can handle up to 200 adjustable parameters and 100 constraints on not adjustable variables as well as upper and lower bounds on all adjustable parameters	Use commercial Solver version (Premium Solver)
			Divide dataset into smaller subsets and fit these independently (usually not recommended)
10	The calculated curve does not approach the experimental data points	Poor starting values for the adjustable parameters	Try different starting values. Ideally, they should be as close as possible to the final values. Qualitatively, the calculated curve should resemble the experimental data already at the beginning of the fitting session
		The regression algorithm is not suitable for the problem at hand	Use different regression algorithm. In the SOLVER OPTIONS window (see **Fig. 5**), try the **Central** option in the **Derivatives** selection. See **Box 4** for details
		The model is poorly scaled	In the SOLVER OPTIONS window (see **Fig. 5**), make sure **Use Automatic Scaling** is checked. See **Box 4** for details
		The model used does not describe the experimental data appropriately	Use appropriate model, keeping in mind that simply adding more adjustable parameters will always result in a better fit. Avoid overfitting (see ANTICIPATED RESULTS for an example)
	The fitted values and the SSR returned by Solver depend on the starting values	Solver gets stuck in local minima	Use starting values close to the expected global minimum

(continued)

TABLE 2 | Troubleshooting table (continued).

Step	Problem	Possible reason	Solution
			Apply a more stringent convergence criterion by reducing the value in the **Convergence** box in the SOLVER OPTIONS window (see **Fig. 5**; lowest possible value is 10^{-300}). See **Box 4** for details
			Use different regression algorithm. In the SOLVER OPTIONS window (see **Fig. 5**), try the **Central** option in the **Derivatives** selection. See **Box 4** for details
		The model is poorly scaled	In the SOLVER OPTIONS window (see **Fig. 5**), make sure **Use Automatic Scaling** is checked. See **Box 4** for details
	The fitted values returned by Solver depend on the starting values, but the SSR is about the same	Overfitting: experimental data are described equally well by different solutions	Carry out confidence analysis, as detailed in Steps 11–18. If the fitted parameters are not defined sufficiently well, fit several datasets simultaneously, apply constraints, reduce the number of adjustable parameters, optimize experimental conditions or use different experimental method (see ANTICIPATED RESULTS)
	The fitted curve resembles the experimental data, but the best-fit values are unrealistic	There is a mistake in the regression equation	Check and fix regression equation. Pay particular attention to units, as these do not show up in spreadsheet formulas
		The model used does not describe the experimental data appropriately	Plot the residuals ($\delta = y_{\mathrm{obs}} - y_{\mathrm{calc}}$) against x as scatter plot. If the residuals are not distributed randomly around zero (i.e., if there are systematic deviations), use a different model. Keep in mind that simply adding more adjustable parameters will always result in a better fit. Avoid overfitting (see ANTICIPATED RESULTS)
14,17	Plot of the SSR versus an adjustable parameter reveals a broad minimum or several minima	Overfitting: experimental data are described equally well by different solutions	Fit several datasets simultaneously, apply constraints, reduce the number of adjustable parameters, optimize experimental conditions or use different experimental method (see ANTICIPATED RESULTS). Do not carry out a detailed calculation of confidence intervals, as done in Steps 15, 16 and 18. See **Box 5** for details
Box 1	In VBA mode: Solver is not listed in the VBA PROJECT REFERENCES window	There is no reference between the Solver add-in and the VBA code	Go to TOOLS→REFERENCES→BROWSE and open Solver in \OFFICE12\LIBRARY\SOLVER (Excel 2003: OFFICE11\LIBRARY\SOLVER)

ANTICIPATED RESULTS

The protocol outlined above is, in principle, applicable to any other dataset or regression equation, provided that the basic prerequisites of NLSF outlined in the INTRODUCTION are met. Of course, the adjustable variables in Steps 3 and 7, the regression equation in Step 5, as well as the value ranges and increments used during confidence assessment in Steps 11–18 will have to be adapted. In any case, it is wise to repeat the fitting procedure several times using different initial values for the adjustable parameters and to subject the best-fit values to careful confidence analysis.

The way replicate measurements (repeated measurements of the dependent variable at the same value of the independent variable) should be considered in a fit depends on whether the replicates are dependent on or independent of one another. For example, two readings on the same solution in a spectrometer are not considered independent, whereas readings on two separately prepared solutions are. In the case of independent measurements, each data point should be included in the fit, whereas dependent measurements should be averaged and only the mean should be included in the fit.

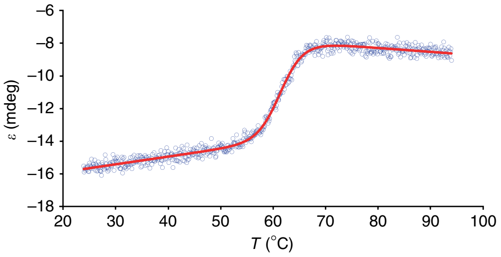


Figure 8 | Circular dichroism (CD) melting curve of 20 μM RNAse A (Sigma–Aldrich, Steinheim, Germany) in 15 mM potassium acetate buffer (pH 5.5). Experimental data (blue scatter) and best fit (red line) determined with the aid of Excel Solver using seven adjustable parameters. The best fit requiring only six fitting parameters is indistinguishable (not shown). ε, ellipticity at 220 nm (given in millidegrees); T, temperature (shown in °C for convenience).

Finally, it cannot be overemphasized that a good fit in terms of a low SSR need not necessarily imply that the model invoked to derive a regression equation and, consequently, the parameter values retrieved from a fitting procedure are correct or make sense. The goodness of fit as expressed by the SSR (see Step 9) and confidence intervals of the adjustable parameters (see Steps 14–18) only make statements about, respectively, the agreement between experimental and fitted data and the reliability with which the best-fit values can be determined from a given dataset. It is of special importance here to note the difference between precision and accuracy. Quoting from a textbook on data reduction and error analysis[4]: "The accuracy of an experiment is a measure of how close the result of the experiment is to the true value; the precision is a measure of how well the result has been determined, without reference to its agreement with the true value."

An example of overfitting: thermal unfolding of a protein

Overfitting is probably the single most important pitfall encountered in data fitting. Overfitting refers to any attempt to extract more parameter values from an experimental dataset than the latter can actually afford. Thus, the fit is said to be underconstrained, or the parameters are underdetermined or redundant. This does not necessarily imply that the experimental data are of poor quality; in most cases, it simply means that the chosen experimental method or setup cannot unambiguously supply the desired number of parameter values. In any case, overfitting causes the values returned for some or all of the adjustable parameters to be imprecise or inaccurate.

This shall be exemplified using the thermal protein unfolding data depicted in **Figure 8** (blue circles). In this experiment, the small globular protein RNAse A was unfolded by raising the temperature from 24 to 94 °C, and loss of secondary structure was monitored by recording the circular dichroism (CD) signal in the far-UV range (see ref. 68 for an experimental protocol). Under equilibrium conditions and making certain assumptions, thermal unfolding of RNAse A and many other proteins can be described by the following regression equation:

$$\varepsilon = \frac{\varepsilon_{\mathrm{u}}^{o} - \varepsilon_{\mathrm{f}}^{o} + (m_{\mathrm{u}} - m_{\mathrm{f}})T}{1 + \exp\left(\frac{1}{R}\left(\Delta C_p \ln\left(\frac{T}{T_{\mathrm{m}}}\right) - \left(\frac{1}{T} - \frac{1}{T_{\mathrm{m}}}\right)(\Delta H_{\mathrm{m}} - T_{\mathrm{m}}\Delta C_p)\right)\right)} + \varepsilon_{\mathrm{f}}^{o} + m_{\mathrm{f}} T$$

Here the independent variable is T, the absolute temperature and the dependent variable is ε, the ellipticity at 220 nm (given in millidegrees). R is the universal gas constant. The adjustable parameters are: T_{m}, the midpoint temperature of

TABLE 3 | Best-fit parameter values returned by Solver on fitting the thermal unfolding curve depicted in **Figure 8** using either six or seven adjustable parameters.

#P	T_{m}	ΔH_{m}	ΔC_{p}	ε_{f}°	ε_{u}°	m_{f}	m_{u}	SSR
	(°C)	(kJ mol^{-1})	(J (mol K)$^{-1}$)	(mdeg)		(mdeg °C^{-1})		(mdeg^2)
6	61.6	427	5	−29.5	−0.24	0.05	−0.02	43.234
7	61.6	427	16	−28.9	−0.95	0.04	−0.02	43.045

ΔC_p, molar isobaric heat capacity change; ε_{f}° and ε_{u}°, ellipticities of, respectively, folded and unfolded protein extrapolated to T = 0 K; ΔH_{m}, molar enthalpy change on unfolding at T_{m}; m_{f} and m_{u}, temperature dependencies of the ellipticities of, respectively, folded and unfolded protein; #P, number of adjustable parameters; SSR, sum of squared residuals; T_{m}, midpoint temperature of thermal unfolding.

thermal unfolding (i.e., the temperature at which half of the protein is unfolded); ΔH_m, the molar enthalpy change on unfolding at T_m; ΔC_p, the (constant) molar isobaric heat capacity change on unfolding; $\varepsilon_\mathrm{f}°$ and $\varepsilon_\mathrm{u}°$, the ellipticities of, respectively, folded and unfolded protein extrapolated to $T = 0$ K; as well as m_f and m_u, the (constant) temperature dependencies of the ellipticities of folded and unfolded protein, respectively. Thus, the regression equation contains seven fitting parameters, three of which (T_m, ΔH_m and ΔC_p) are thermodynamic parameters, whereas the other four ($\varepsilon_\mathrm{f}°$, $\varepsilon_\mathrm{u}°$, m_f and m_u) are required to represent the dependencies of pre- and post-transition baselines on temperature. A discussion of the assumptions made in deriving the above regression equation is beyond the scope of this protocol but can be found elsewhere[68]. Irrespective of this, it is obvious that fitting a simple sigmoidal curve with sloping pre- and post-transition baselines using seven adjustable parameters is prone to serious overfitting.

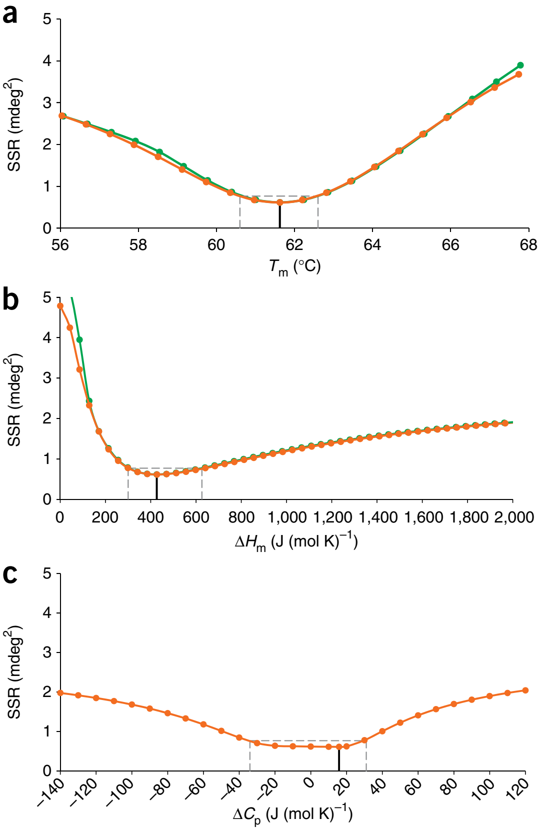


Figure 9 | Confidence assessment of best-fit parameter values. Plots of sum of squared residuals (SSR) against (**a**) midpoint temperature (T_m), (**b**) molar enthalpy change (ΔH_m) and (**c**) molar isobaric heat capacity change (ΔC_p). Orange lines and symbols refer to a seven-parameter fit (with adjustable ΔC_p), whereas green lines and symbols refer to a six-parameter fit (with ΔC_p fixed at 5 J (mol K)$^{-1}$). Dashed lines mark parameter values at which the SSR amounts to 125% of its minimal value. Vertical black lines indicate best-fit values.

As illustrated in **Figure 8**, the best fit (red line) based on seven adjustable parameters is excellent and returns the values summarized in **Table 3**. However, **Figure 9** reveals that the confidence of the best-fit values of some parameters is, at best, mediocre (orange lines and symbols). T_m displays the highest confidence among the three parameters (**Fig. 9a**), but a plot of the SSR against ΔH_m is strongly asymmetric (**Fig. 9b**), revealing that the SSR is extremely insensitive to unrealistically positive deviations from the optimal ΔH_m value. Finally, a diagram depicting the dependence of the SSR on ΔC_p (**Fig. 9c**) is characterized by a broad trough ranging from -20 J (mol K)$^{-1}$ to $+20$ J (mol K)$^{-1}$ in which the SSR is almost indifferent to ΔC_p. Negative values are particularly suspect because unfolding of globular proteins is usually accompanied by an increase in heat capacity. In the specific case of RNAse A, direct determination of the heat capacity change by differential scanning calorimetry has yielded a value of $\Delta C_p = 5$ J (mol K)$^{-1}$ (ref. 69). The considerable deviation between this and the best-fit value of $\Delta C_p = 16$ J (mol K)$^{-1}$ might be surprising at first glance, but inspection of **Figure 9c** reveals that the difference in the SSR is negligible and that the tremendously poor confidence of this parameter is because of overfitting.

It is obvious that the complex, asymmetric shapes obtained on plotting the SSR versus ΔH_m or ΔC_p cannot be captured by a single parameter, as might be implied by fitting programs reporting a best-fit value ± a standard error or standard deviation. To illustrate this point, we analyzed the same dataset using the commercial fitting program Origin. Like most other commercially available software packages for NLSF, this program calculates the so-called asymptotic standard errors. Confidence intervals can then be assigned by multiplying these standard errors by an appropriate constant taken from statistical tables[5]. Although Origin determined the same best-fit values as found by Solver, the 95% confidence intervals calculated by Origin were only ±0.1 °C for T_m, ±24 kJ mol^{-1} for ΔH_m and ±5 J (mol K)$^{-1}$ for ΔC_p, which clearly are gross underestimations of the real confidence intervals. Also, derivation of asymmetric confidence intervals as outlined above in Steps 15, 16 and 18 is not applicable (see **Box 5** for details). In the absence of such well-established tools for confidence assessment, lower and upper parameter estimates can be obtained by arbitrarily defining a threshold SSR value. In the present example, the dashed lines in **Figure 9** delimit the parameter ranges within which the SSR does not exceed 125% of the minimal value (as suggested in ref. 63). Lower and upper parameter estimates retrieved from such a procedure may help in quantifying the qualitative statements of the previous paragraph, but it should be kept in mind that the SSR threshold was chosen arbitrarily and that no single quantity can account for the complex shape of the SSR curves.

The problem of overfitting can be minimally alleviated by fixing the ill-behaved parameter at $\Delta C_p = 5$ J (mol K)$^{-1}$, thus decreasing the number of fitting parameters to six and slightly improving the confidence of the other adjustable parameters (green lines and symbols in **Fig. 9a,b**). However, the goodness of the best fit in terms of the SSR is virtually unaffected because both the best-fit and the independently determined ΔC_p values lie in the SSR trough in **Figure 9c**. In this particular case, further enhancement could be achieved by a number of remedies: (i) the temperature could be scanned over a greater range (e.g., 0–100 °C rather than 24–94 °C). This would give greater confidence to the four parameters defining the pre- and post-transition baselines, such that the latter could compensate less readily for variations in the thermodynamic parameters. (ii) Several experiments carried out under identical conditions might be fitted simultaneously (globally). (iii) Experiments could be carried out under various conditions (e.g., different pH values or in the presence of stabilizing or destabilizing additives like urea or trimethylamine N-oxide, respectively), and the data could be fitted simultaneously. This would require an additional term in the regression equation accounting for the effect of the varied parameter (i.e., a linkage between the influence of temperature and the influence of pH or additive). However, as the baselines would be the same for all datasets, the number of adjustable parameters per dataset would be reduced (A. Sieber and S. Keller, unpublished data). (iv) In addition to ΔC_p, either T_m or ΔH_m might be determined using an independent method, and the remaining thermodynamic parameter could be extracted from CD unfolding curves with greater confidence.

ACKNOWLEDGMENTS We thank Professor Heiko Heerklotz (University of Toronto, Canada), Dr. Alekos Tsamaloukas (Rensselaer Polytechnic Institute, USA), Natalie Bordag (FMP) and Martin Kemmer (23karat GmbH, Berlin, Germany) for helpful discussions and constructive comments on the manuscript. This work was supported by grant KE 1478/1-1 from the Deutsche Forschungsgemeinschaft (DFG) to S.K.

AUTHOR CONTRIBUTIONS G.K. designed and performed experiments, analyzed and fitted data, and wrote the manuscript. S.K. designed experiments, analyzed data and wrote the manuscript.

Published online at http://www.natureprotocols.com.
Reprints and permissions information is available online at http://npg.nature.com/reprintsandpermissions.

1. De Levie, R. *Advanced Excel for Scientific Data Analysis* 2nd edn. (Oxford University Press, New York, 2008).
2. Johnson, M.L. Why, when, and how biochemists should use least squares. *Anal. Biochem.* **206**, 215–225 (1992).
3. Press, W.H., Teukolsky, A.S., Vetterling, W.T. & Flannery, B.P. Modeling of data. In *Numerical Recipes in C: The Art of Scientific Computing* 2nd edn. 656–706 (Cambridge University Press, New York, 1992).
4. Bevington, P.R. & Robinson, D.K. Least-squares fit to an arbitrary function. In *Data Reduction and Error Analysis for the Physical Sciences* 3rd edn. 142–167 (McGraw-Hill Higher Education, New York, 2009).
5. Motulsky, H. & Christopoulos, A. *Fitting Models to Biological Data Using Linear and Nonlinear Regression: A Practical Guide to Curve Fitting* 2nd edn. (GraphPad Software, San Diego, 2003).
6. Orvis, W.J. *Excel for Scientists and Engineers* 1st edn. (SYBEX, Alameda, 1995).
7. Fylstra, D., Lasdon, L., Watson, J. & Waren, A. Design and use of the Microsoft Excel Solver. *Interfaces* **28**, 29–55 (1998).
8. Lasdon, L.S., Waren, A.D., Jain, A. & Ratner, M. Design and testing of a generalized reduced gradient code for nonlinear programming. *ACM T. Math. Software* **4**, 34–50 (1987).
9. Beck, A., Tsamaloukas, A.D., Jurcevic, P. & Heerklotz, H. Additive action of two or more solutes on lipid membranes. *Langmuir* **24**, 8833–8840 (2008).
10. Tsamaloukas, A.D., Beck, A. & Heerklotz, H. Modeling the micellization behavior of mixed and pure *n*-alkyl-maltosides. *Langmuir* **25**, 4393–4401 (2009).
11. Plouffe, L. Jr. & Luxenberg, S.N. Biological modeling on a microcomputer using standard spreadsheet and equation solver programs: the hypothalamic-pituitary-ovarian axis as an example. *Comput. Biomed. Res.* **25**, 117–130 (1992).
12. Hargrove, J.L., Heinz, G. & Heinz, O. Modeling transitions in body composition: the approach to steady state for anthropometric measures and physiological functions in the Minnesota human starvation study. *Dyn. Med.* **7**, 16 (2008).
13. Stevens, P.W. & Kelso, D.M. Estimation of the protein-binding capacity of microplate wells using sequential ELISAs. *J. Immunol. Methods* **178**, 59–70 (1995).
14. Kawamata, W. & Toyoshima, H. Estimation of T1 and T2 using general-purpose spreadsheet software. *Nippon Hoshasen Gijutsu Gakkai Zasshi* **65**, 306–311 (2009).
15. Rohatagi, S., Hochhaus, G., Möllmann, H., Barth, J. & Derendorf, H. Pharmacokinetic interaction between endogenous cortisol and exogenous corticosteroids. *Pharmazie* **50**, 610–613 (1995).
16. Dansirikul, C., Choi, M. & Duffull, S.B. Estimation of pharmacokinetic parameters from non-compartmental variables using Microsoft Excel. *Comput. Biol. Med.* **35**, 389–403 (2005).
17. Meineke, I. & Brockmöller, J. Simulation of complex pharmacokinetic models in Microsoft Excel. *Comput. Methods Programs Biomed.* **88**, 239–245 (2007).
18. Briones, A.M. Jr. & Reichardt, W. Estimating microbial population counts by 'most probable number' using Microsoft Excel. *J. Microbiol. Methods* **35**, 157–161 (1999).
19. Sonnenberg, A. Special review: game theory to analyse management options in gastro-oesophageal reflux disease. *Aliment. Pharmacol. Ther.* **14**, 1411–1417 (2000).
20. Ward, R., Schlenker, J. & Anderson, G.S. Simple method for developing percentile growth curves for height and weight. *Am. J. Phys. Anthropol.* **116**, 246–250 (2001).
21. Zhang, F. & Roush, W.B. Multiple-objective (goal) programming model for feed formulation: an example for reducing nutrient variation. *Poult. Sci.* **81**, 182–192 (2002).
22. Guevara, V.R. Use of nonlinear programming to optimize performance response to energy density in broiler feed formulation. *Poult. Sci.* **83**, 147–151 (2004).
23. Kuo, P.C., Schroeder, R.A., Mahaffey, S. & Bollinger, R.R. Optimization of operating room allocation using linear programming techniques. *J. Am. Coll. Surg.* **197**, 889–895 (2003).
24. Maurer, M., Kühleitner, M., Gasser, B. & Mattanovich, D. Versatile modeling and optimization of fed batch processes for the production of secreted heterologous proteins with *Pichia pastoris*. *Microb. Cell Fact.* **5**, 37 (2006).
25. Abdel-Fattah, Y.R. *et al.* Application of factorial experimental designs for optimization of cyclosporin A production by *Tolypocladium inflatum* in submerged culture. *J. Mol. Microbiol. Biotechnol.* **17**, 1930–1936 (2007).
26. Burke, J.A. Two mathematical programming models of cheese manufacture. *J. Dairy Sci.* **89**, 799–809 (2006).
27. Schrader, H. & Svec, A. Comparison of ionization chamber efficiencies for activity measurements. *Appl. Radiat. Isot.* **60**, 369–378 (2004).
28. Brown, A.M. A step-by-step guide to non-linear regression analysis of experimental data using a Microsoft Excel spreadsheet. *Comput. Methods Programs Biomed.* **65**, 191–200 (2001).
29. Brown, A.M. A non-linear regression analysis program for describing electrophysiological data with multiple functions using Microsoft Excel. *Comput. Methods Programs Biomed.* **82**, 51–57 (2006).
30. Branco, T.J., Botelho do Rego, A.M., Ferreira, M.I. & Vieira Ferreira, L.F. Luminescence lifetime distributions analysis in heterogeneous systems by the use of Excel's Solver. *J. Phys. Chem. B* **109**, 15958–15967 (2005).
31. Li, J. Comparison of the capability of peak functions in describing real chromatographic peaks. *J. Chromatogr. A* **952**, 63–70 (2002).
32. Nikitas, P., Pappa-Louisi, A. & Papageorgiou, A. On the equations describing chromatographic peaks and the problem of the deconvolution of overlapped peaks. *J. Chromatogr. A* **912**, 13–29 (2001).
33. Nikitas, P., Pappa-Louisi, A., Papageorgiou, A. & Zitrou, A. On the use of genetic algorithms for response surface modeling in high-performance liquid chromatography and their combination with the Microsoft Solver. *J. Chromatogr. A* **942**, 93–105 (2002).
34. Karmarkar, S., Garber, R., Kluza, J. & Koberda, M. Gel permeation chromatography of dextrans in parenteral solutions: calibration procedure development and method validation. *J. Pharm. Biomed. Anal.* **41**, 1260–1267 (2006).

35. Dasgupta, P.K. Chromatographic peak resolution using Microsoft Excel Solver. The merit of time shifting input arrays. *J. Chromatogr. A* **1213**, 50–55 (2008).
36. van Dijk, J.W. Thermoluminescence glow curve deconvolution and its statistical analysis using the flexibility of spreadsheet programs. *Radiat. Prot. Dosimetry* **119**, 332–338 (2006).
37. Walsh, S. & Diamond, D. Non-linear curve fitting using Microsoft Excel Solver. *Talanta* **42**, 561–572 (1995).
38. Kane, P. & Diamond, D. Determination of ion-selective electrode characteristics by non-linear curve fitting. *Talanta* **44**, 1847–1858 (1997).
39. Luther, G.W. III, Theberge, S.M. & Rickard, D. Determination of stability constants for metal-ligand complexes using the voltammetric oxidation wave of the anion/ligand and the DeFord and Hume formalism. *Talanta* **51**, 11–20 (2000).
40. Comuzzi, C., Polese, P., Melchior, A., Portanova, R. & Tolazzi, M. SOLVERSTAT: a new utility for multipurpose analysis. An application to the investigation of dioxygenated Co(II) complex formation in dimethylsulfoxide solution. *Talanta* **59**, 67–80 (2003).
41. Safavi, A., Maleki, N., Rostamzadeh, A. & Maesum, S. CCD camera full range pH sensor array. *Talanta* **71**, 498–501 (2007).
42. Parsons, J.D. A high-throughput method for fitting dose–response curves using Microsoft Excel. *Anal. Biochem.* **360**, 309–311 (2007).
43. Bárány-Wallje, E. *et al.* A critical reassessment of penetratin translocation across lipid membranes. *Biophys. J.* **89**, 2513–2521 (2005).
44. Keller, S., Böthe, M., Bienert, M., Dathe, M. & Blume, A. A simple fluorescence-spectroscopic membrane translocation assay. *ChemBioChem* **8**, 546–552 (2007).
45. Keller, S., Tsamaloukas, A. & Heerklotz, H. A quantitative model describing the selective solubilization of membrane domains. *J. Am. Chem. Soc.* **127**, 11469–11476 (2005).
46. Schmidt, M.F., El-Dahshan, A., Keller, S. & Rademann, J. Selective identification of cooperatively binding fragments in a high-throughput ligation assay enables the evolution of a picomolar caspase-3 inhibitor. *Angew. Chem. Int. Ed.* **48**, 6346–6349 (2009).
47. Keller, S. *et al.* Membrane-mimetic nanocarriers formed by a dipalmitoylated cell-penetrating peptide. *Angew. Chem. Int. Ed.* **44**, 5252–5255 (2005).
48. Keller, S., Heerklotz, H., Jahnke, N. & Blume, A. Thermodynamics of lipid membrane solubilization by sodium dodecyl sulfate. *Biophys. J.* **90**, 4509–4521 (2006).
49. Heerklotz, H., Tsamaloukas, A.D. & Keller, S. Monitoring detergent-mediated solubilization and reconstitution of lipid membranes by isothermal titration calorimetry. *Nat. Protoc.* **4**, 686–697 (2009).
50. Keller, S., Heerklotz, H. & Blume, A. Monitoring lipid membrane translocation of sodium dodecyl sulfate by isothermal titration calorimetry. *J. Am. Chem. Soc.* **128**, 1279–1286 (2006).
51. Geissler, D. *et al.* (Coumarin-4-yl)methyl esters as highly efficient, ultrafast phototriggers for protons and their application to acidifying membrane surfaces. *Angew. Chem. Int. Ed.* **44**, 1195–1198 (2005).
52. Hagen, V. *et al.* Coumarinylmethyl esters for ultrafast release of high concentrations of cyclic nucleotides upon one- and two-photon photolysis. *Angew. Chem. Int. Ed.* **44**, 7887–7891 (2005).
53. Cambridge, S.B., Geissler, D., Keller, S. & Cürten, B. A caged doxycycline analogue for photoactivated gene expression. *Angew. Chem. Int. Ed.* **45**, 2229–2231 (2006).
54. Gilbert, D. *et al.* Caged capsaicins: new tools for the examination of TRPV1 channels in somatosensory neurons. *ChemBioChem* **8**, 89–97 (2007).
55. Sauer, I. *et al.* Dipalmitoylation of a cellular uptake-mediating apolipoprotein E-derived peptide as a promising modification for stable anchorage in liposomal drug carriers. *Biochim. Biophys. Acta.* **1758**, 552–561 (2006).
56. Tsamaloukas, A.D., Keller, S. & Heerklotz, H. Uptake and release protocol for assessing membrane binding and permeation by way of isothermal titration calorimetry. *Nat. Protoc.* **2**, 695–704 (2007).
57. Seelig, J. Titration calorimetry of lipid–peptide interactions. *Biochim. Biophys. Acta.* **1331**, 103–116 (1997).
58. Seelig, J. Thermodynamics of lipid–peptide interactions. *Biochim. Biophys. Acta.* **1666**, 40–50 (2004).
59. Forrest, S. Genetic algorithms: principles of natural selection applied to computation. *Science* **261**, 872–878 (1993).
60. Motulsky, H.J. & Ransnas, L.A. Fitting curves to data using nonlinear regression: a practical and nonmathematical review. *FASEB J.* **1**, 365–374 (1987).
61. Johnson, M.L. Evaluation and propagation of confidence intervals in nonlinear, asymmetrical variance spaces. Analysis of ligand-binding data. *Biophys. J.* **44**, 101–106 (1983).
62. Johnson, M.L. & Frasier, S.G. Nonlinear least-squares analysis. *Methods Enzymol.* **117**, 301–342 (1985).
63. Johnson, K.A., Simpson, Z.B. & Blom, T. FitSpace Explorer: an algorithm to evaluate multidimensional parameter space in fitting kinetic data. *Anal. Biochem.* **387**, 30–41 (2009).
64. Michaelis, L. & Menten, M.L. Die Kinetik der Invertinwirkung. *Biochem. Z.* **49**, 333–369 (1913).
65. Lineweaver, H. & Burk, D. The determination of enzyme dissociation constants. *J. Am. Chem. Soc.* **56**, 658–666 (1934).
66. Berg, J.M., Tymoczko, J.L. & Stryer, L. *Biochemistry* 5th edn. (W.H. Freeman & Company, New York, 2002).
67. Wisniak, J. & Polishuk, A. Analysis of residuals—a useful tool for phase equilibrium data analysis. *Fluid Phase Equilib.* **164**, 61–82 (1999).
68. Greenfield, N.J. Using circular dichroism collected as a function of temperature to determine the thermodynamics of protein unfolding and binding interactions. *Nat. Protoc.* **1**, 2527–2535 (2006).
69. Pace, C.N., Grimsley, G.R., Thomas, S.T. & Makhatadze, G.I. Heat capacity change for ribonuclease A folding. *Protein Sci.* **8**, 1500–1504 (1999).