Curve Fitting Made Easy

Scientists and engineers often want to represent empirical data using a model based on mathematical equations. With the correct model and calculus, one can determine important characteristics of the data, such as the rate of change anywhere on the curve (first derivative), the local minimum and maximum points of the function (zeros of the first derivative), and the area under the curve (integral). The goal of data (or curve) fitting is to find the parameter values that most closely match the data. The models to which data are fitted depend on adjustable parameters.

Take the formula

$$Y = A^*\exp(-X/X_0)$$

in which X is the independent variable, Y is the dependent variable, and A and X_0 are the parameters. In an experiment, one typically measures variable Y at a discrete set of values of variable X. Curve fitting provides two general uses for this formula:

- A person believes the formula describes the data and wants to know what parameter values of A and X_0 correspond to the data.
- A person wants to know which of several formulas best describes the data.

To perform fitting, we define some function—which depends on the parameters—that measures the closeness between the data and the model. This function is then minimized to the smallest possible value with respect to the parameters. The parameter values that minimize the function are the best-fitting parameters. In some cases, the parameters are the coefficients of the terms of the model, for example, if the model is polynomial or Gaussian. In the simplest case, known as linear regression, a straight line is fitted to the data. In most scientific and engineering models, however, the dependent variable depends on the parameters in a nonlinear way.

Model selection

One needs a good understanding of the underlying physics, chemistry, electronics, and other properties of a problem to choose the right model. No graphing or analysis software can pick a model for the given data—it can only help in differentiating between models. Once a model is picked, one can do a rough assessment of it by plotting the data. At least some agreement is needed between the data and the model curve. If the model is supposed to represent exponential growth and the data are monotonically decreasing, the model is obviously wrong. Many commercial software programs perform linear and nonlinear regression analysis and curve fitting using the most common models. The better programs also allow users to specify their own fitting functions.

A typical example of exponential decay in physics is radioactive decay. The measurable rate of decay as a function of time—that is, the number of decays per unit of time—is given by

$$\Lambda(t) = \Lambda(0)e^{-\lambda t}$$

where t is elapsed time and λ is the probability of decay of one nucleus during one time unit. By measuring the decay rate as a function of time and fitting the data to the exponential-decay law, one can infer the constant λ .

As another example, the following Gaussian function describes a normal distribution:

$$y\ (x) = \frac{1}{\sigma\sqrt{2\pi}}\ \exp\left[-\frac{(x-x_0)^2}{2\sigma^2}\right]$$

Here, x_0 is the mean value and σ is the standard deviation of the distribution. Usually, several measurements are performed of a quantity x. The results are binned—assigned to discrete intervals called bins—as a function of x, such that $y(x)$ represents the number of counts of the values in the bin centered around x. The mean value and standard deviation are then obtained by curve fitting.

Sometimes, it is possible to linearize a nonlinear model. For example, consider the first equation. If we take the logarithm of both the left- and the righthand side, we get

$$\log(Y) = \log(A) - X/X_0$$

Hence, $\log(Y)$ is a linear function of X. This means that one could try to fit the logarithm of Y data to a linear function of X data to find A and X_0. This approach was widely used before the age of computers. There are problems with it, however, including the fact that the transformed data usually do not satisfy the assumptions of linear regression. Linear regression assumes that the scatter of points around the line follows a Gaussian distribution, and that the standard deviation is the same at every value of the independent variable.

To find the values of the model's parameters that yield the curve closest to the data points, one must define a function that measures the closeness between the data and the model. This function depends on the method used to do the fitting, and the function is typically chosen as the sum of the squares of the vertical deviations from each data point to the curve. (The deviations are first squared and then added up to avoid cancellations between positive and negative values.) This method assumes that the measurement errors are independent and normally distributed with constant standard deviation.

Linear regression

The simplest form of modeling is linear regression, which is the prediction of one variable from another when the relationship between them is assumed to be linear. One variable is considered to be independent and the other dependent.

A linear regression line has an equation of the form

$$y = a + bx,$$

where x is the independent variable and y is the dependent variable. The slope of the line is b, and a is the intercept (the value of y when $x = 0$). The goal is to find the values of a and b that make the line closest to the data. The chi-square estimator in this case (where σ_i are measurement errors; if they are not known, they should all be set to 1) is given by

$$\chi^2(a,b) = \sum_{i=1}^{N}(\frac{y_i - a - bx_i}{\sigma_i})^2$$

χ^2 is a function of the parameters of the model; N is the total number of points. We need to find the values of a and b that minimize χ^2. Typically, this means differentiating χ^2 with respect to the parameters a and b and setting the derivatives equal to 0. For a linear model, the minimum can be found and expressed in analytic form as

$$0 = \frac{\partial \chi^2}{\partial a} = -2\sum_{i=1}^{N} \frac{y_i - a - bx_i}{\sigma_i^2}$$

$$0 = \frac{\partial \chi^2}{\partial b} = -2\sum_{i=1}^{N} \frac{x_i\left(y_i - a - bx_i\right)}{\sigma_i^2}$$

The resulting system of two equations with two unknowns can be solved explicitly for a and b, producing the regression line like that shown in Figure 1.

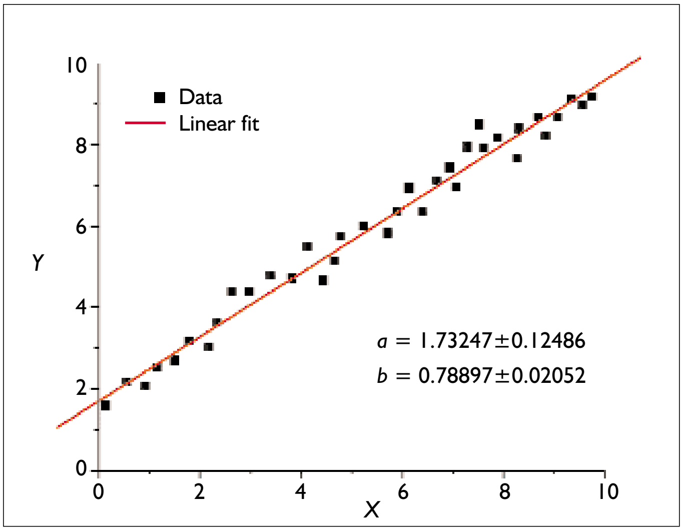

Figure 1. In linear regression, the correct values of *a* and *b* in the straight line equation are those that minimize the value of χ^2.

Nonlinear models

In most cases, the relationship between measured values and measurement variables is nonlinear. Nonlinear curve fitting also seeks to find those parameter values that minimize the deviations between the observed y values and the expected y values. In nonlinear models, however, one usually cannot solve the equation for the parameters. Various iterative procedures are used instead. Nonlinear fitting always starts with an initial guess of parameter values. One usually looks at the general shape of the model curve and compares it with the shape of the data points. For example, look at the sigmoidal function in Figure 2.

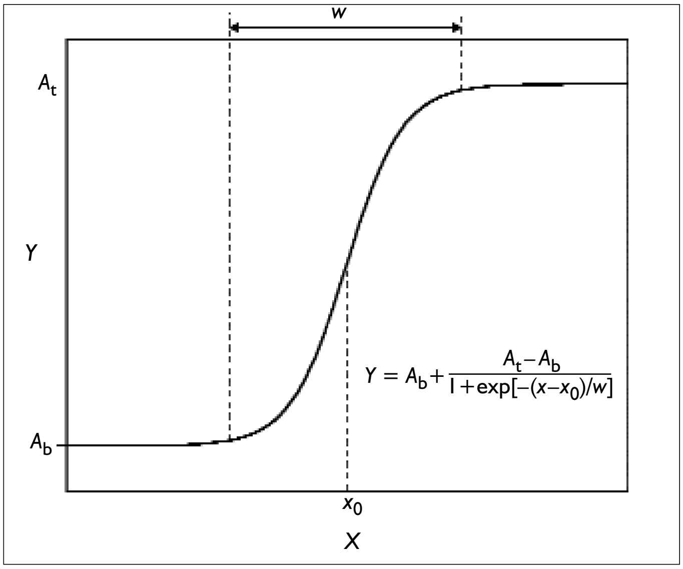

Figure 2. In nonlinear curve-fitting, such as this sigmoidal function, to minimize the deviations between the observed and expected y values, it is necessary to estimate parameter values and use iterative procedures.

This function has four parameters. A_b and A_t are two asymptotic values—values that the function approaches but never quite reaches—at small and large x. The curve crosses over between these two asymptotic values in a region of x values whose approximate width is w and which is centered around x_0. For the initial estimate of the values of A_b and A_t, we can use the average of the lower third and the upper third of the Y values of data points. The initial value of x_0 is the average of X values of the crossover area, and the initial value of w is the width of the region of points between the two asymptotic values.

A good understanding of the selected model is important because the initial guess of the parameter values must make sense in the real world. In addition, a good fitting software will try to provide an initial guess for the functions built into it. The nonlinear fitting procedure is iterative. The most widely used iterative procedure for nonlinear curve fitting is the *Levenberg–Marquardt algorithm*, which good curve-fitting softwares implement. Starting from some initial values of parameters, the method tries to minimize

$$\chi^2(a,b) = \sum_{i=1}^{N}\left[\frac{y_i - f(x_i,a,b)}{\sigma_i}\right]^2$$

by successive small variations of parameter values and reevaluation of χ^2 until a minimum is reached. The function f is the one to which we are trying to fit the data, and χ^2 and y_i are the X and Y values of data points. Although the method involves sophisticated mathematics, end users of typical curve-fitting software usually only need to initialize parameters and press a button to start the iterative fitting procedure.

Weighting

Often, curve fitting is done without weighting. In the above formulas for χ^2, the weights σ_i are all set to 1. If experimental measurement errors are known, they should be used as σ_i. They then provide an uneven weighting of different experimental points: the smaller the measurement error for a particular experimental point, the larger the weight of that point in χ^2. Proper curve-fitting software allows users to supply measurement errors for use in the fitting process. If the experimental errors are not known exactly, it is sometimes reasonable to make an assumption. For example, if you believe that the deviation of the experimental data from the theoretical curve is proportional to the value of the data, then you can use

$$\sigma_i = y_i$$

Here, a weight of each point, which is the inverse of the square of σ_i, is inversely proportional to the square of the value of a data point.

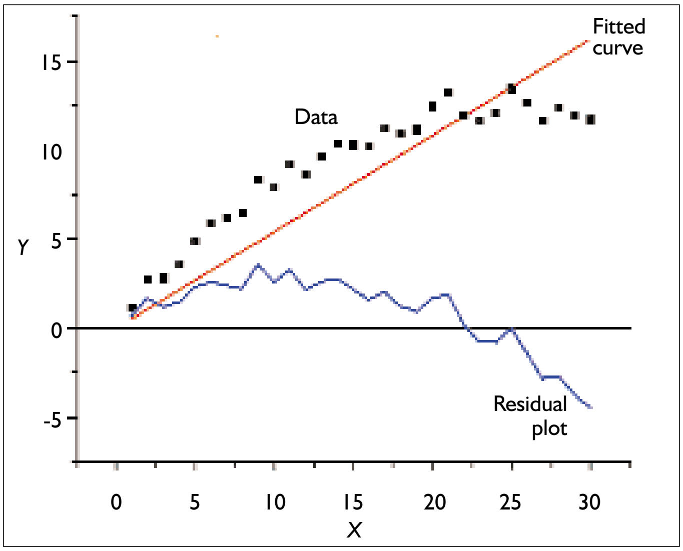

Figure 3. The residual plot (blue) shows the deviations of the data points (black) from the fitted curve (red) and indicates by its nonrandomness a possible hidden variable.

Interpreting results

In assessing modeling results, we must first see whether the program can fit the model to the data—that is, whether the iteration converged to a set of values for the parameters. If it did not, we probably made an error, such as choosing the wrong model or initial values. A look at the graphed data points and fitted curve can show if the curve is close to the data. If it is not, then the fit has failed, perhaps because the iterative procedure did not find a minimum of χ^2 or the wrong model was chosen.

The parameter values must also make sense from the standpoint of the model. If, for example, one of the parameters is temperature in kelvins and the value obtained from fitting is negative, we have obtained a physically meaningless result. This is true even if the curve seemingly does fit the data well. In this example, we have likely chosen an incorrect model.

There is a quantitative value that describes how well the model fits the data. The value is defined as χ^2 divided by the number of degrees of freedom, and it is referred to as the *standard deviation of the residuals*, which are the vertical distances of the data points from the line. The smaller this value, the better the fit.

An even better quantitative value is R^2, known as the *goodness of fit*. It is computed as the fraction of the total variation of the Y values of data points that is attributable to the assumed model curve, and its values typically range from 0 to 1. Values close to 1 indicate a good fit.

To assess the certainty of the best-fit parameter values, we look at the obtained *standard errors of parameters*, which indirectly tell us how well the curve fits the data. If the standard error is small, it means that a small variation in the parameter would produce a curve that fits the data much worse. Therefore, we say that we know the value of that parameter well. If the standard error is large, a relatively large variation of that parameter would not spoil the fit much; therefore, we do not really know the parameter's value well.

Important intervals

Standard errors are related to another quantity, one usually obtained from the fitting and computed by curve-fitting software: the *confidence interval*. It says how good the obtained estimate of the value of the fitting function is at a particular value of X. The confidence interval gives a desired level of statistical confidence, expressed as a percentage, usually 95% or 99%. For a particular value of X, we can claim that the correct value for the fitting function lies within the obtained confidence interval. Good curve-fitting software computes the confidence interval and draws the confidence interval lines on the graph with the fitted curve and the data points. The smaller the confidence interval, the more likely the obtained curve is close to the real curve.

Another useful quantity usually produced by curve-fitting software is the *prediction interval*. Whereas the confidence interval deals with the estimate of the fitting function, the prediction interval deals with the data. It is computed with respect to a desired confidence level p, whose values are usually chosen to be between 95% and 99%. The prediction interval is the interval of Y values for a given X value within which p% of all experimental points in a series of repeated measurements are expected to fall. The narrower the interval, the better the predictive nature of the model used. The prediction interval is represented by two curves lying on opposite sides of the fitted curve.

The *residuals plot* provides insight into the quality of the fitted curve. Residuals are the deviation of the data points from the fitted curve. Perhaps the most important information obtained is whether there are any *lurking (hidden) variables*. In Figure 3, the data points are black squares, the obtained fitted line is red, and the blue line represents residuals.

Normally, if the curve fits the data well, one expects the data-point deviations to be randomly distributed around the fitted curve. But in this graph, for the data points whose values are less than about 22, all residuals are positive; that is, the data points are above the fitted curve. However, for data points with *X* values above 22, all residuals are negative and the data points fall below the fitted curve. This indicates a nonrandom behavior of residuals, which means that this behavior may be driven by a hidden variable.

So curve fitting, although it cannot tell you how, can certainly tell you when your chosen model needs improvement.

Further reading

1. www.originlab.com (Origin).

2. http://mathworld.wolfram.com/LeastSquaresFitting.html

3. Press, W. H.; et al. *Numerical Recipes in C*, 2nd ed.; Cambridge University Press: New York, 1992; 994 pp.; provides a detailed description of the Levenberg–Marquardt algorithm.

4. Bevington, P. R. *Data Reduction and Error Analysis for the Physical Sciences*, McGraw-Hill, New York, 1969.

5. Draper, N. R.; Smith, H. *Applied Regression Analysis*, 3rd edition, John Wiley & Sons, 1998. Ω

BIOGRAPHY

Marko Ledvij is a senior software developer at OriginLab Corp. in Framingham, Massachusetts (marko@originlab.com).