Tutorials and worked examples for simulation, curve fitting, statistical analysis, and plotting.
http://www.simfit.org.uk

Simulating a data set that is exact to computer precision is the first step in simulation, so that random error can be added retrospectively to simulate experimental results.

Choosing the [A/Z] option from the SIMFIT main menu is used to open the program **makdat**, which then allows you to create such almost-exact data from a library of models, or from a user-supplied model.

Before attempting a simulation it is essential to understand these issues.

1. Choosing the correct model
2. Choosing sensible parameters
3. Choosing a reasonable range for the independent variable(s)
4. Choosing a meaningful technique for the distribution of evaluation points
5. Viewing the current simulation
6. Saving the simulated data to a file

Program **makdat** has access to the particular library that is contained in either `w_models.dll` for 32-bit applications, or `x64_models.dll` for 64-bit applications, and there are numerous versions of these where the standard library has been augmented for special user requirements. The standard version has models for one, two, and three variables as well as for single differential equations. For systems of differential equations program **deqsol** must be used.

1. Choosing the correct model

It is essential that users should have a good idea of what would be an appropriate mathematical model.

For instance, it is assumed that biochemists would know which of the following ligand binding models should be used for simulating data for two binding sites.

$$f(x) = \frac{A_1K_1x}{1 + K_1x} + \frac{A_2K_2x}{1 + K_2x}$$

$$g(x) = \frac{\beta}{2}\left\{\frac{\phi_1x + 2\phi_1\phi_2x^2}{1 + \phi_1x + \phi_1\phi_2x^2}\right\}$$

Again, chemists would be expected to know which of the following double exponential models to use for reactants in the two linked irreversible chemical reaction scheme.

$$F(t) = \left\{\frac{p_1p_3}{p_2 - p_1}\right\}\exp(-p_1t) + \left\{p_4 - \frac{p_1p_3}{p_2 - p_1}\right\}\exp(-p_2t)$$

$$G(t) = p_3 + p_4 + p_5 - \left\{\frac{p_2p_3}{p_2 - p_1}\right\}\exp(-p_1t) - \left\{p_4 - \frac{p_1p_3}{p_2 - p_1}\right\}\exp(-p_2t)$$

So the first step in simulation is to choose the correct model and appreciate the relationship between the library parameters, which are all in the form $p_1, p_2, \ldots, p_n$, and the more usual dedicated nomenclature such as $K_m, V_{max}, k_{app}, A_0, B_0$ and so on.

Note that the models contained in the basic library are defined in the SIMFIT reference manual `w_manual.pdf`.

2. Choosing sensible parameters

Most users will have a good idea of the parameter values required, normally because these are values reported from curve-fitting and it is wished to check the fit to data based on data simulated using the best-fit parameters.

Often some idea of parameter values can be obtained by simply inspecting experimental data. For instance, all the parameters in models $f(x), g(x), F(t)$, and $G(t)$ must be nonnegative. The final asymptotic level reached by $f(x)$ and $g(x)$ as $x \to \infty$ are of course $A_1 + A_2$ and β respectively, while the binding constants will tend to be around the value of x at the half saturation point. Again, model $F(t)$ tends to zero while $G(t)$ tends to $q_3 + q_4 + q_5$ as $t \to \infty$ and the exponential parameters will be of the order of the half life since $F(t)$ is the intermediate species $B(t)$ while $G(t)$ is the final product $C(t)$ in the irreversible consecutive reaction

$$A \overset{k_1}{\to} B \overset{k_2}{\to} C$$

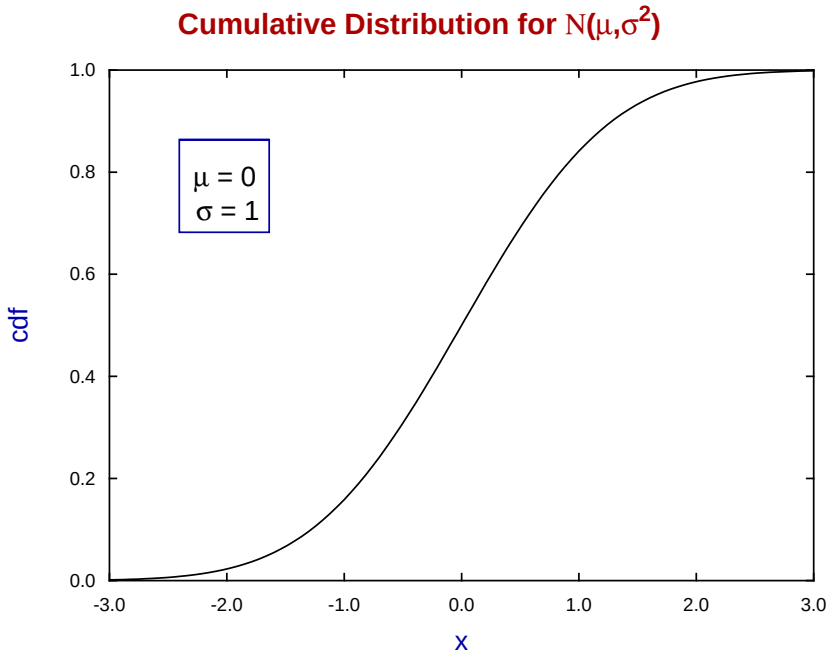
so that $p_1 = k_1, p_2 = k_2, p_3 = A(0), p_4 = B(0)$, and $p_5 = C(0)$.

3. Choosing a reasonable range for the independent variable(s)

In the case of functions of one variable there are four distinct ways to choose starting points and end points that define the range for the independent variable.

- User inputs starting and ending X values manually.
- User inputs starting and ending Y values and corresponding X values are calculated numerically.
- User reads a set of values from a file like `vector.tf1`.
- User edits (or simply accepts) the current data.

For an example of how to set the range manually, run the program **makdat**, select functions of one variable, pick statistical distributions, choose the normal *cdf*, decide to have a zero constant term, set the mean $p(1) = 0$, fix the standard deviation $p(2) = 1$, input the scaling factor $p(3) = 1$ and then choose $X_{\text{start}} = -3$ and $X_{\text{stop}} = 3$ to generate the next figure (after adding the maths).



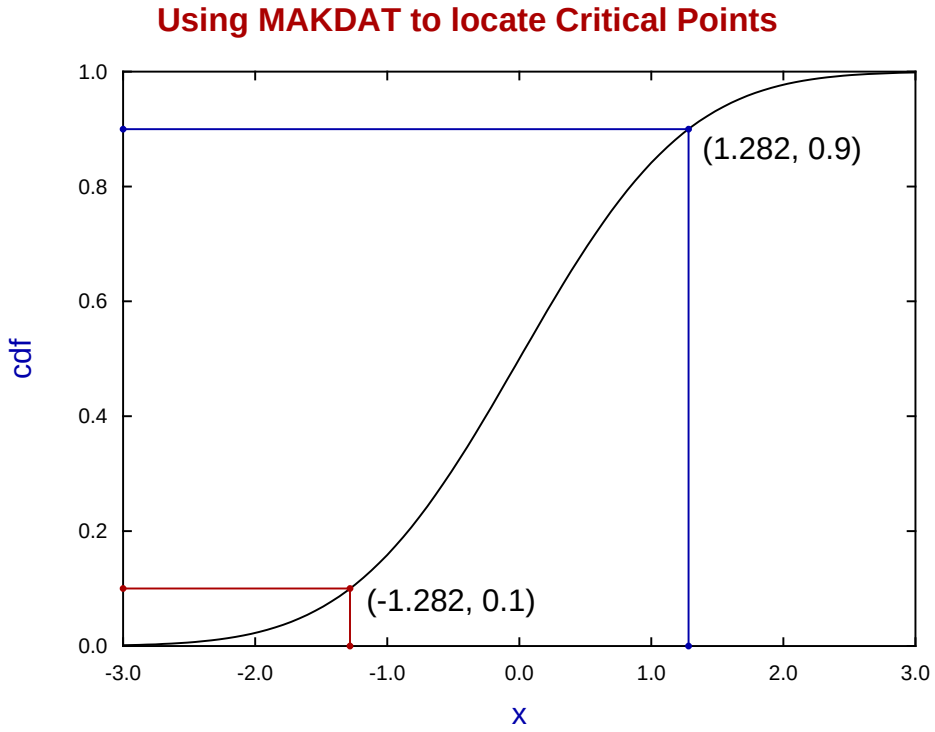
The process of finding a range of x for simulation when this depends on fixed values of y, that is to find $x = x(y)$ when there is no simple explicit expression for $x(y)$, is frequently required. For instance, finding $x = x(y)$ when $y(x)$ is the normal distribution function

$$y(x) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{x} \exp -\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2 dx$$

where $\sigma > 0$. After setting parameters $\mu = 0, \sigma = 1$, it is clear from the previous graph that, to find the point x_1 where $y(x_1) = 0.1$ it would be reasonable to guess $-3 \le x_1 \le -1$, while to locate the point x_2 where $y(x_2) = 0.9$ it would be sensible to guess that $1 \le x_2 \le 3$. Using these initial estimates program **makdat** refined them to give the following results

$$X_{\text{start}} = -3,\ X_{\text{stop}} = -1: \quad y_1 = 0.1,\ x_1 = -1.282,$$
$$X_{\text{start}} = 1,\ X_{\text{stop}} = 3: \quad y_2 = 0.9,\ x_2 = 1.282.$$

These critical points are illustrated by the next figure.



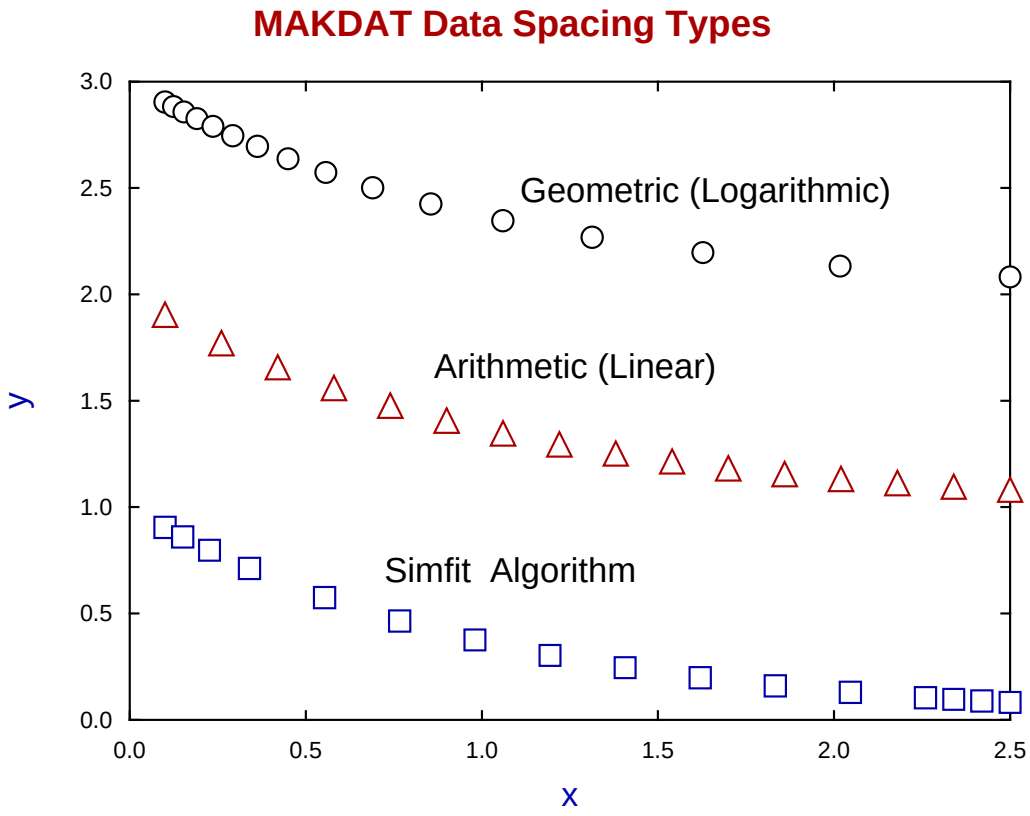
Note that, when attempting such root-finding calculations, **makdat** will attempt to alter the starting estimates by decreasing X_{start} and increasing X_{stop} if a root cannot be located, but it will not change the sign of these starting estimates. In the event of problems locating roots, there is no substitute for plotting the function to get some idea of the position of the roots, as shown in the previous figures.

4. Choosing a meaningful technique for the distribution of evaluation points

For functions of one variable program **makdat** provides three ways to space points.

- Points in a geometric progression (logarithmic spacing)
- Points in an arithmetic progression (linear spacing)
- Points following a SIMFIT spacing algorithm

To illustrate these data spacing options consider the next figure.



Geometric (Logarithmic) spacing

This spacing leads to consecutive points in a geometric progression, i.e. with increasing separation between consecutive points, and requires positive coordinates. From the point of view of optimum design for parameter estimation and model discrimination with models like the exponentials and rational functions so often encountered in experiments, this is the best choice as discussed in the following publication.

Optimal design for model discrimination using the F test with non-linear biochemical models. Criteria for choosing the number and spacing of experimental points.
Bardsley,W.G., McGinlay,P.B & Roig,M.G. (1989) *J. theor. Biol.* **139**, 85-102

Arithmetic (Linear) spacing

This spacing leads to consecutive points in an arithmetic progression, i.e. with a constant separation between consecutive points, and is the design most used, probably out of of convenience. However it is a bad choice for many situations where it is best to place more points where model equations change most rapidly, i.e. near the origin.

Simfit Algorithm

The SIMFIT spacing is used by SIMFIT when a best-fit curve is calculated for plotting against experimental data. This spacing is a compromise as it attempts to cover the cases where data and best-fit curves are plotted in other transformations than the logarithmic, like a Scatchard plot or double reciprocal plot for instance. It approximates to geometric spacing for early points and reverse geometric spacing for late points.

5. Viewing the current simulation

Once a data set has been simulated it is always possible to display a table of the independent variables and simulated values. In addition, for one independent variable a plot can be created, while for two independent variables a surface can be plotted together with contours if required. It is not possible to plot a function of three independent variables.

6. Saving the simulated data to a file

Files saved will have the independent variable(s) in the first column(s), then the simulated values in the penultimate column, followed by a final column of weights.

There are three reasons for saving the simulated data to a file.

- **Using the saved file for fitting**

  It is assumed that normally data are simulated in order to be fitted. So a final column of weights (usually 5% of the calculated function value) will be added to the saved file to make it a curve fitting file. The last column must be left in for weighting, or else replaced by a column of 1 if unweighted fitting is required. This is easily done using program **editmt**.

- **Using the saved file for plotting**

  As the final column of weights is not necessary for plotting it would be usual to delete the last column of weights using program **editmt**. However, for plotting functions of one variable using program **simplot** there is no need to delete the last column (i.e. column 3). On the other hand, for a functions of two or three variables for plotting, the last column of weights must always be deleted.

- **Adding random error**

  As the data written to file will be almost exact, they can be used to confirm that a model simulated with no added error can be fitted to return the correct best-fit parameters. If it is wished to add random error to simulate experimental data, the files can be input into program **adderr**, which can also overwrite the default weights by alternative weighting schemes if required.