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

It is convenient to deal separately with univariate calibration and bioassay, as multivariate calibration is performed in SIMFIT using the partial least squares (PLS) technique.

Calibration

This requires fitting a curve $y = f(x)$ to a (x, y) training data set with x known exactly and y measured with limited error, so that the best fit model $\hat{f}(x)$ can then be used to predict x_i given arbitrary y_i. Usually the model is of no significance and steps are taken to use a data range over which the model is approximately linear, or at worst a shallow smooth curve. It is assumed that experimental errors arising when constructing the best fit curve are uncorrelated and normally distributed with zero mean, so that the standard curve is a good approximation to the maximum likelihood estimate.

- **Calibration curves**

  Creating and using a standard calibration curve involves:

  1. Measuring responses y_i at fixed values of x_i, and using replicates to estimate s_i, the sample standard deviation of y_i if possible.
  2. Preparing a curve fitting type file with x, y, and s using program **makfil**, and using **makmat** to prepare a vector type data file with x_i values to predict y_i.
  3. Finding a best fit curve $y = f(x)$ to minimize $WSSQ$, the sum of weighted squared residuals.
  4. Supplying y_i values and predicting x_i together with 95% confidence limits, i.e. inverse-prediction of $x_i = \hat{f}^{-1}(y_i)$. Sometimes you may also need to evaluate $y_i = \hat{f}(x_i)$.

  It may be that the s_i are known independently, but often they are supposed constant and unweighted regression, i.e. all $s_i = 1$, is unjustifiably used. Any deterministic model can be used for $f(x)$, e.g., a sum of logistics or Michaelis-Menten functions using program **qnfit**, but this could be unwise. Calibration curves arise from the operation of numerous effects and cannot usually be described by one simple equation. Use of such equations can lead to biased predictions and is not always recommended. Polynomials are useful for gentle curves as long as the degree is reasonably low (≤ 3 ?) but, for many purposes, a weighted least squares data smoothing cubic spline is the best choice. Unfortunately polynomials and splines are too flexible and follow outliers, leading to oscillating curves, rather than the data smoothing that is really required. Also they cannot fit horizontal asymptotes. You can help in several ways.

  a) Get good data with more distinct x-values rather than extra replicates.

  b) If the data approach horizontal asymptotes, either leave some data out as they are no use for prediction anyway, or try using $\log(x)$ rather than x, which can be done automatically by program **calcurve**.

  c) Experiment with the weighting schemes, polynomial degrees, spline knots or constraints to find the optimum combinations for your problem.

  d) Remember that predicted confidence limits also depend on the s values you supply, so either get the weighting scheme right, or set all all $s_i = 1$.

- **Turning points in calibration curves**

  Some programs will warn you if $f(x)$ has a turning point, since this can make inverse prediction ambiguous. You can then re-fit to get a new curve, eliminate bad data points, get new data, etc., or

carry on if the feature seems to be harmless. You will be given the option of searching upwards or downwards for prediction in such ambiguous cases. It should be obvious from the graph, nature of the mathematical function fitted, or position of the turning point in which direction the search should proceed.

- **Calibration using polynomials**

  For linear or almost linear data you can use program **linfit** which just fits straight lines of the form

  $$f(x) = p_0 + p_1 x.$$

  However, for smooth gentle curves, program **polnom** is preferred because it can also fit a polynomial

  $$f(x) = p_0 + p_1 x + p_2 x^2 + \cdots + p_n x^n,$$

  where the degree n is chosen according to statistical principles. What happens is that **polnom** fits all polynomials from degree 0 up to degree 6 and gives statistics necessary to choose the statistically justified best fit n. However, in the case of calibration curves, it is not advisable to use a value of n greater than 2 or at most 3, and warnings are issued if the best fit standard curve has any turning points that could make inverse prediction non-unique.

- **Calibration using cubic splines**

  If a polynomial of degree 2 or at most 3 is not adequate, a cubic spline calibration curve could be considered. It does not matter how nonlinear your data are, **calcurve** can fit them with splines with several types of knots and tension. The best-fit spline curve from programs such as **calcurve**, **compare**, and **spline** can be archived for repeated initialization of a reference standard curve to use for calibration.

- **Advanced calibration using special models**

  Sometimes you would want to use a specific mathematical model for calibration such as a straight line through the origin, or a quadratic with no linear term, but other models might be more appropriate. For instance, a mixture of two High/Low affinity binding sites or a cooperative binding model might be required for a saturation curve, or a mixture of two logistics might adequately fit growth data. If you know an appropriate model for the standard curve, use **qnfit** for inverse prediction because, after fitting, the best-fit curve can be used for calibration, or for estimating derivatives or areas under curves (AUC) if appropriate.

Bioassay

This is a special type of calibration, where the data are obtained over as wide a range as possible, nonlinearity is accepted (e.g. a sigmoid curve), and specific parameters of the underlying response, such as the time to half-maximum response, final size, maximum rate, area AUC, $EC50$, $LD50$, or $IC50$ are to be estimated. With bioassay, a known deterministic model may be required, and assuming normally distributed errors may sometimes be a reasonable assumption, but alternatively the data may consist of proportions in one of two categories (e.g. alive or dead) as a function of some treatment, so that binomial error is more appropriate and probit analysis, or similar, is called for.

A special type of inverse prediction is required when equations are fitted to dose response data in order to estimate some characteristic parameter, such as the half time $t_{1/2}$, the area under the curve AUC, or median effective dose in bioassay (e.g. $ED50$, $EC50$, $IC50$, $LD50$, etc.), along with standard errors and 95% confidence limits. The model equations used in this sort of analysis are not supposed to be exact models constructed according to scientific laws, rather they are empirical equations, selected to have a shape that is close to the shape expected of such data sets. So, while it is is pedantic to insist on using a model based on scientific model building, it is important to select a model that fits closely over a wide variety of conditions.

Older techniques, such as using data subjected to a logarithmic transform in order to fit a linear model, are no longer called for as they are very unreliable, leading to biased parameter estimates. Hence, in what follows, it is assumed that data are to be analyzed in standard, not logarithmically transformed coordinates, but there is nothing to prevent data being plotted in transformed space after analysis, as is frequently done when the independent variable is a concentration, i.e., it is desired to have an the independent variable proportional to chemical potential. The type of analysis called for depends very much on the nature of the data, the error distribution involved, and the goodness of fit of the assumed model. It is essential that data are obtained over a wide range, and that the best fit curves are plotted and seen to be free from bias which could seriously degrade routine estimates of percentiles, say. The only way to decide which of the following procedures should be selected for your data, is to analyze the data using those candidate models that are possibilities, and then to adopt the model that seems to perform best, i.e., gives the closest best fit curves and most sensible inverse predictions.

- **Exponential models**

  If the data are in the form of a simple or multiphasic exponential decline from a finite value at $t = 0$ to zero as $t \to \infty$, and half times $t_{1/2}$, or areas AUC are required, use **exfit** to fit one or a sum of two exponentials with no constant term.

- **Trapezoidal estimation**

  If no deterministic model can be used for the AUC it is usual to prefer the trapezoidal method with no data smoothing, where replicates are simply replaced by means values that are then joined up sequentially by sectional straight lines. The program **average** is well suited to this sort of analysis.

- **The Hill equation**

  This empirical equation is

$$f(x) = \frac{Ax^n}{B^n + x^n},$$

  which can be fitted using program **inrate**, with either n estimated or n fixed, and it is often used in sigmoidal form (i.e. $n > 1$) to estimate the maximum value A and half saturation point B, with sigmoidal data (not data that are only sigmoidal when x-semilog transformed, as all binding isotherms are sigmoidal in x-semilog space).

- **Ligand binding and enzyme kinetic models**

  There are three cases:
  a) data are increasing as a function of an effector, i.e., ligand or substrate, and the median effective ligand concentration $ED50$ or apparent $K_m = EC50 = ED50$ is required,
  b) data are a decreasing function of an inhibitor $[I]$ at fixed substrate concentration $[S]$ and $IC50$, the concentration of inhibitor giving half maximal inhibition, is required, or
  c) the flux of labeled substrate $[Hot]$, say, is measured as a decreasing function of unlabeled isotope $[Cold]$, say, with $[Hot]$ held fixed.
  If the data are for an increasing saturation curve and ligand binding models are required, then **hlfit** or, if cooperative effects are present, **sffit** can be used to fit one or two binding site models.

  More often, however, an enzyme kinetic model, such as the Michaelis-Menten equation will be used as now described. To estimate the maximum rate and apparent K_m, i.e., $EC50$ the equation fitted by **mmfit** in substrate mode would be

$$v([S]) = \frac{V_{max}[S]}{K_m + [S]}$$

  while the interpretation of $IC50$ for a reversible inhibitor at concentration $[i]$ with substrate fixed at concentration S would depend on the model assumed as follows.

$$\text{Competitive inhibition } v([I]) = \frac{V_{max}[S]}{K_m(1 + I/K_i) + [S]}$$
$$IC50 = \frac{K_i(K_m + [S])}{K_m}$$
$$\text{Uncompetitive inhibition } v([I]) = \frac{V_{max}[S]}{K_m + [S](1 + [I]/K_i]}$$
$$IC50 = \frac{K_i(K_m + [S])}{[S]}$$
$$\text{Noncompetitive inhibition } v([I]) = \frac{V_{max}[S]}{(1 + [I]/K_i)(K_m + [S])}$$
$$IC50 = K_i$$
$$\text{Mixed inhibition } v([I]) = \frac{V_{max}[S]}{K(1 + [I]/K_{i1}) + [S](1 + [I]/K_{i2})}$$
$$IC50 = \frac{K_{i1}K_{i2}(K_m + [S])}{(K_m K_{i2} + [S]K_{i1})}$$
$$\text{Isotope displacement } v([Cold]) = \frac{V_{max}[Hot]}{K_m + [Hot] + [Cold]}$$
$$IC50 = K_m + [Hot]$$

Of course, only two independent parameters can be estimated with these models, and, if higher order models are required and justified by statistics and graphical deconvolution, the apparent V_{max} and apparent K_m are then estimated numerically.

- **Growth curves**

  If the data are in the form of sigmoidal increase, and maximum size, maximum growth rate, minimum growth rate, $t_{1/2}$ time to half maximum size, etc. are required, then use **gcfit** in growth curve mode. For instance, with the logistic model

$$f(t) = \frac{A}{1 + B\exp(-kt)}$$
$$t_{1/2} = \frac{\log(B)}{k}$$

  the maximum size A and time to reach half maximal size $t_{/2}$ are estimated.

- **Survival curves**

  If the data are independent estimates of fractions remaining as a function of time or some effector, i.e. sigmoidally decreasing profiles fitted by **gcfit** in mode 2, and $t_{1/2}$ is required, then normalize the data to proportions of time zero values and use **gcfit** in survival curve mode 2. The Weibull model

$$S(t) = 1 - \exp(-(At)^B)$$
$$t_{1/2} = \frac{\log(2)}{AB}.$$

  is very useful.

- **Survival time models**

  If the data are in the form of times to failure, possibly censored, then **gcfit** should be used in survival time mode 3. With survival time models the median survival time $t_{1/2}$ is estimated, where

$$\int_0^{t_{1/2}} f_T(t)\,dt = \frac{1}{2},$$

  and $f_T(t)$ is the survival probability density function.

- **Models for proportions**

  If the data are in the form of numbers of successes (or failures) in groups of known size as a function of some control variable and you wish to estimate percentiles, e.g., $EC50$, $IC50$, or maybe $LD50$ (the median dose for survival in toxicity tests), use **gcfit** in GLM dose response mode. This is because the error distribution is binomial, so generalized linear models should be used. You should

95% confidence regions in inverse prediction

polnom estimates non-symmetrical confidence limits assuming that the N values of y for inverse prediction and weights supplied for weighting are exact, and that the model fitted has n parameters that are justified statistically. **calcurve** uses the weights supplied, or the estimated coefficient of variation, to fit confidence envelope splines either side of the best fit spline, by employing an empirical technique developed by simulation studies. Root finding is employed to locate the intersection of the y_i supplied with the envelopes. The AUC, LD50, half-saturation, asymptote and other inverse predictions in SIMFIT use a t distribution with $N-n$ degrees of freedom, and the variance-covariance matrix estimated from the regression. That is, assuming a prediction parameter defined by $p = f(\theta_1, \theta_2, \ldots, \theta_n)$, a central 95% confidence region is constructed using the prediction parameter variance estimated by the propagation of errors formula

$$\hat{V}(p) = \sum_{i=1}^{n} \left(\frac{\partial f}{\partial \theta_i} \right)^2 \hat{V}(\theta_i) + 2 \sum_{i=2}^{n} \sum_{j=1}^{i-1} \frac{\partial f}{\partial \theta_i} \frac{\partial f}{\partial \theta_j} \hat{CV}(\theta_i, \theta_j).$$

Note that this formula for the propagation of errors can be used to calculate parameter standard errors for parameters that are calculated as functions of parameters that have been estimated by fitting, such as apparent maximal velocity when fitting sums of Michaelis-Menten functions. However, such estimated standard errors will only be very approximate.