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

The Von Bertalanffy differential equation arose in allometric studies in order to represent growth as the result of a balance between anabolic and catabolic processes.

For instance, a spherical bacterium of radius r has a rate of absorbtion of nutrients proportional to the surface area $4\pi r^2$ but a rate of metabolism proportional to the volume $4\pi r^3/3$, so the ratio of surface area to bulk decreases as $3/r$ and it would be anticipated that the rate of change in size y as a function of time x, say dy/dx, would increase rapidly for small y but slow down as y increases. This leads to the expression

$$\frac{dy}{dx} = p_1 y^{p_2} - p_3 y^{p_4}, \quad y(0) = p_5$$

for some exponents p_2 and p_4 with $p_4 > p_2 \geq 0$, where to model a growth process y and the parameters p_i would be nonnegative. For certain special parameter values formal integral is possible and leads to some of the classical growth equations as follows.

$$\begin{aligned}
\text{Exponential model } dS/dt &= kS \\
S(t) &= A\exp(kt), \text{ where } A = S_0 \\
\text{Monomolecular model } dS/dt &= k(A - S) \\
S(t) &= A[1 - B\exp(-kt)], \text{ where } B = 1 - S_0/A \\
\text{Logistic model } dS/dt &= kS(A - S)/A \\
S(t) &= A/[1 + B\exp(-kt)], \text{ where B } = A/S_0 - 1 \\
\text{Von Bertalanffy 2/3 model } dS/dt &= \eta S^{2/3} - \kappa S \\
S(t) &= [A^{1/3} - B\exp(-kt)]^3, \\
&\text{where } A^{1/3} = \eta/\kappa, B = \eta/\kappa - S_0^{1/3}, k = \kappa/3
\end{aligned}$$

However the Von Bertalanffy differential equation will be used to illustrate the methods available using SimFIT to study the properties of a differential equation, followed by simulation and curve fitting.

To examine this differential equation requires choosing the fixed parameters and the initial condition so the following values will be used for this purpose.

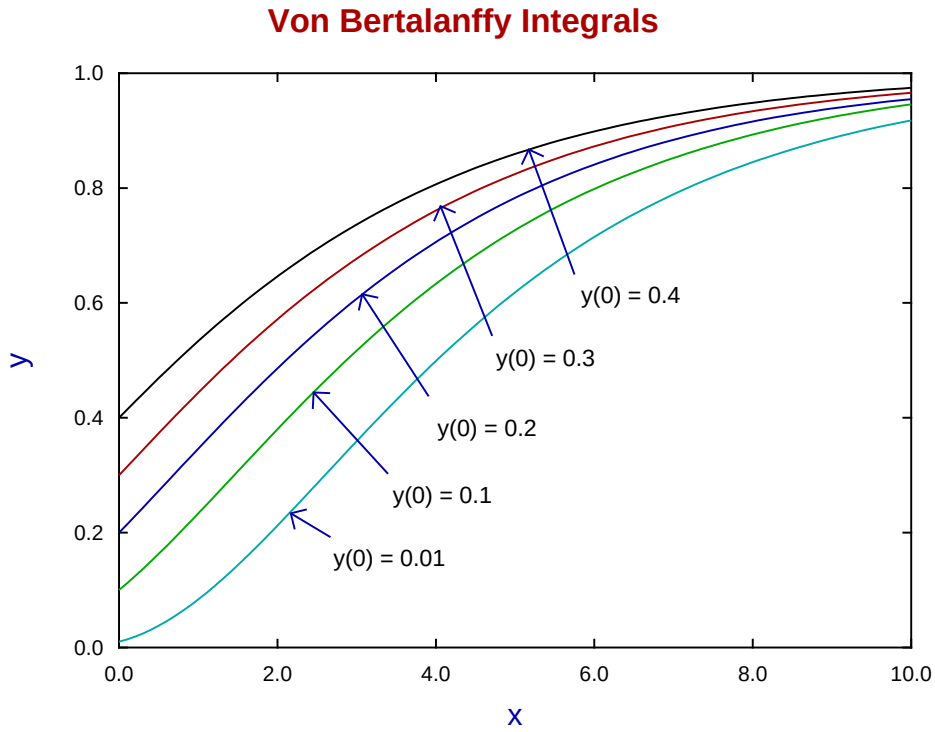
$$\begin{aligned}
p_1 &= 1 \\
p_2 &= 2/3 \\
p_3 &= 1 \\
p_4 &= 1 \\
p_5 &= 0.01
\end{aligned}$$

Of course this is the original case where a formal integral exists and it is an interesting exercise to consider the alternative $y(x)$ profiles that result as these parameters are varied. However, with these parameter values the following observations can be made.

1. The slope is zero when y is zero and when y is 1
2. As x does not appear on the right hand side the slope is fixed given any y for $0 \leq y \leq 1$
3. The $y(x)$ curve is monotonically increasing for $x > 0$

Initial conditions

Here, for example, is a series of plots illustrating how increasing the initial condition $y(0)$ from $p_5 = 0.01$ to $p_5 = 0.4$ simply slides the profiles to the left.



To make composite plots of $y(x)$ profiles such as this for any differential equations use the following protocol.

1. Open program **deqsol** from the [A/Z] option on the main SIMFIT menu and select the differential equation required from the library of pre-compiled models, or by reading in a user-defined model.
2. Edit the parameters as required, which in this example using the library model is just p_5.
3. Plot the $y(x)$ data between the end points selected (in this case 100 points for $0 \leq x \leq 10$) and use the [Advanced] option to save the coordinates to file. Note that, after doing this, the files saved can be added to your graph plotting archive for retrospective plotting as part of a group.
4. Input the group of coordinate files into program **simplot** either individually, from your project archive, or best of all from a library file created using program **maklib**, then edit as required.

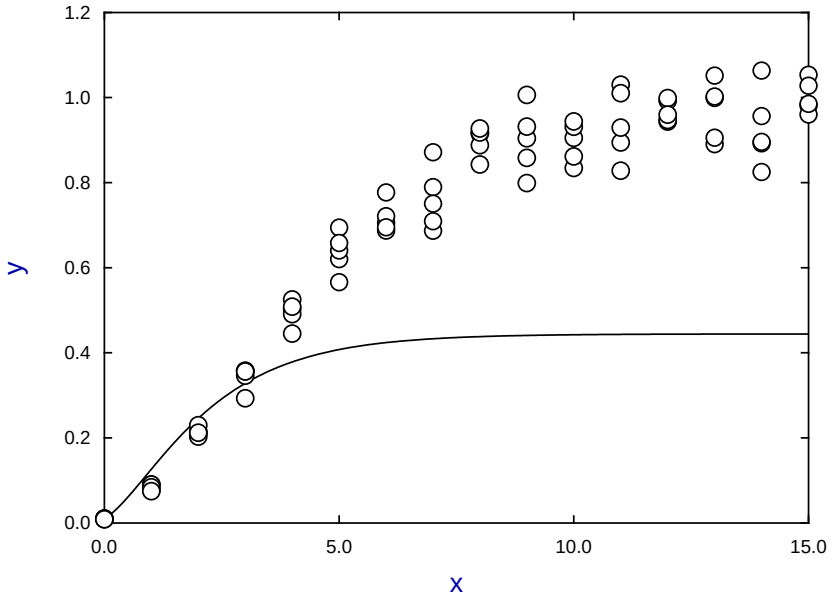
Simulating and fitting

As we are dealing with a single differential equation, it is best to proceed as follows

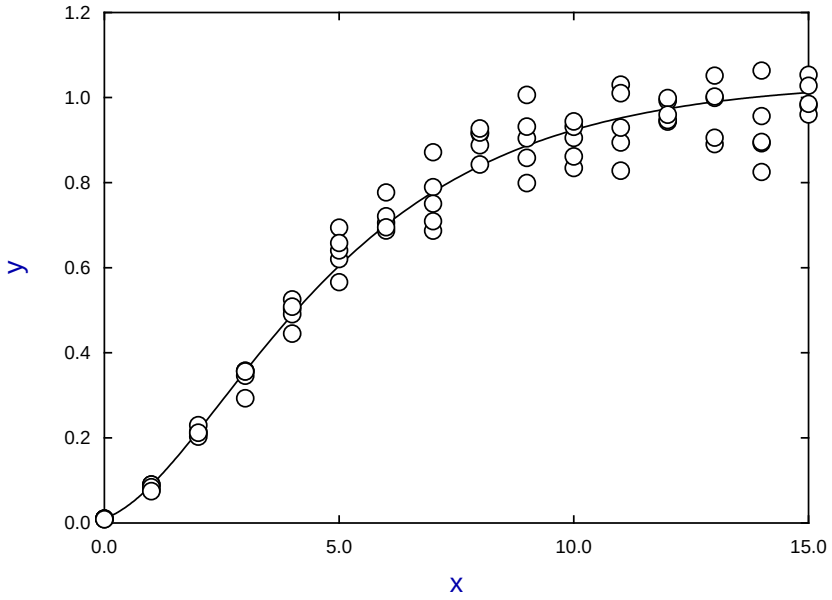
- Use program **makdat** to simulate a $y(x)$ profile.
- Use program **adderr** to add random error.
- Finally fit using program **qnfit**.

That is because these programs provide more options than are available with program **deqsol**, which is really designed to work with systems of differential equations. Using program **qnfit** to fit the default data set `qnfit_ode.tf2` results in the following outcome. First the overlay before fitting, then the overlay after fitting, and finally the results.

Data and Starting-Estimate-Curve



Data and best-fit curve



Best-fit parameters for curve-fit 1 using LBFGSB/DVODE

Number	Low-Limit	High-Limit	Value	Std. Error	Lower95%cl	Upper95%cl	*p*
1	0.50	1.50	0.887160	0.0029641	0.88126	0.89306	0.0000
2	0.25	0.75	0.616683	0.0009505	0.61479	0.61858	0.0000
3	0.50	1.50	0.875717	0.0021387	0.87146	0.87998	0.0000
4	0.75	1.25	0.943496	0.0026182	0.93828	0.94871	0.0000
5	0.0001	0.05	0.010133	0.0000739	0.00999	0.00103	0.0000

Of course this is an artificial data set over an extended range using a model defined in the SIMFIT library, and the fitting included the initial condition for purposes of illustration, which should be avoided at all costs with actual data.

The von Bertalannffy user defined model

For information, the corresponding default user-defined model file is `deqmod1_e.tf6` which is now listed.

```
%
  model: von Bertalannfy growth model

 differential equation: f(1) = dy(1)/dx
                             = p(1)*y(1)^p(2) - p(3)*y(1)^p(4)

              Jacobian: j(1) = df(1)/dy(1)
                             = p(1)*p(2)*y(1)^(p(2) - 1.0)
                               -p(3)*p(4) y(1)^(p(4) - 1.0)

    initial condition: y0(1) = p(5)
 %
 1 equation
 differential equation
 5 parameters
 %
 begin{expression}
 f(1) = p(1)y(1)^p(2) - p(3)y(1)^p(4)
 end{expression}
 %
 begin{expression}
 A = p(2) - 1.0
 B = p(4) - 1.0
 j(1) = p(1)p(2)y(1)^A - p(3)p(4)y(1)^B
 end{expression}
 %

Coding for the model is now finished but parameter starting values,
curve fitting limits, and range of integration can be appended.
If these are not supplied DEQSOL will need an initialisation file.

 begin{limits}
 0  1.0       3
 0  0.6666667 3
 0  1.0       3
 0  1.0       3
 0  0.01      1
 end{limits}

 begin{range}
 121
 0
 10
 end{range}
```
