Facit, omdugga 2

William Lövfors

2017-03-02

1 Constructing a model

1.1 ODEs and reaction rates

$$
\begin{aligned}
d/dt([A]) &= -v_1 + v_3 \\
d/dt([B]) &= v_1 - v_4 - v_2 \\
d/dt([C]) &= v_2 - v_3 \\
[A](0) &= 0 \\
[B](0) &= 0 \\
[C](0) &= 100
\end{aligned}
$$

$$
\begin{aligned}
v_1 &= k_1 \cdot [A] \cdot u \\
v_2 &= k_2 \cdot [B] \\
v_3 &= k_3 \cdot [C] \\
v_4 &= k_4 \cdot [B]
\end{aligned}
$$

where $k_1 = 1$, $k_2 = 2$, $k_3 = 3$, $k_4 = 4$, and $u = 1$.
Note, values of k_1 to k_4 and initial values are arbitrarily chosen.

1.2 Measurement equation

$$\hat{y} = k \cdot [B]$$

where $k = 5$ (arbitrarily chosen).

2 Simulating the model

2.1 Euler forward

In Euler's forward method, the current state of the system is used to estimate the next state using the current derivatives, and taking a small step forward.

The derivatives are then recalculated and another step is taken forward. The algorithm (with arbitrary values inserted) can be summarized as the following:

1. Collect initial values ($y(0) = 0$).
   Calculate initial derivatives ($y' = d/dt(y)$)

2. Take a step (h), calculate new states, $y_{n+1} = y(n \cdot h) = y_n + h \cdot y'$.
   Assuming $n = 0, y_0 = 0, y' = 2, h = 1$: $y_1 = y(1) = 0 + 1 \cdot 2 = 2$

3. If $t_n = n \cdot h = t_{end}$ stop, else set $n = n + 1$, calculate y' and repeat step 2.

Note, t_{end} is the desired end time.

2.2 Letting t tend to infinity

As can be seen in the interaction graph (and the equations), B can either be recycled through C and A, or it can be degraded. Since there is no inflow to the system, B will decrease over time. As t increases, B is constantly removed from the system. When $t \to \infty$, $B \to 0$.

2.3 Changing v_2 dynamics

From before:

$$v_2 = k_2 \cdot [B] \tag{1}$$

To change equation 1 into a Michaelis-Menten reaction, we need to add a dependency on $[B]$. The new equation will be as follows:

$$v_2 = k_{max} \frac{[B]}{k_m + [B]} \tag{2}$$

where $k_{max} = 10, k_m = 4$ (arbitrary values).

3 The cost function

$$V(p) = \sum_{t=0}^{n} \left(\frac{(y_t - \hat{y}(p)_t)^2}{SEM_t^2} \right) \tag{3}$$

where $V(p)$ is the quantified agreement between data (y, SEM) and simulation $(\hat{y}(p))$ depending on the parameters (p).

3.1 input/output

Input to the cost-function is the parameters (and structure) of the model (and the data). The output is the agreement with data. In essence, the function simulates the model and calculates the cost according to equation 3

3.2 Residuals

The residuals are the distances between the model simulation and the experimental data ($y - \hat{y}$). They are weighted by the uncertainty in data, and summarized, see equation 3. As they are only the distances between the simulation and experimental data, they can be negative. However, when they are summarized into the cost, the sign is removed (in our case by raising to the power of 2).

3.3 What does the cost represent?

See equation 3. The cost represents the distances between the model simulation and the experimental data, normalized by the uncertainty of data, summarized over all time points. Another way to express the same thing, is that the cost represents how well the model is able to explain the experimental data.

4 Optimization and statistics

4.1 Why do we use a cost-function?

For the optimization to work, we need to have a value representing how well the current set of parameters make the simulation behave like the experimental data. If we did not have this, we would not be able to know if a change is an improvement or not. The optimization algorithms attempts to minimize the cost.

4.2 Having a good start guess

When using this algorithm, an initial guess is necessary. The algorithm will then attempt to tune the parameters, to make the agreement between data and simulation as good as possible. Since we are limited by computational power, it is not feasible to sample the whole parameter space, therefore we are limited in the amount of parameter sets to test. By having a good start guess, the chance of reaching the global minimum is increased.

4.3 Two models passing a χ^2-test

If we have two models passing a χ^2-test, we are not able to reject either of the model (assuming all available data are used in the cost function). This means that we have two candidates for the true model structure. We should now look for predictions where the models deviate, and do more experiments.

5 Predictions and experiments

5.1 Incorrect input signals

In this case, we can still use our results. Assuming we have 1 μM as input strength, we can give the model 1 or 1000, depending on if we want go express the signal in μM or nM (or any other unit). 1 μM = 1000 nM. However, this will change how one should interpret the parameters in the model. If we express everything in μM, the rate constants (parameters) will also be expressed in μM/time-unit (time-unit can also vary between seconds, hours, days etc). Since it does not matter in what unit we express the signal, we can still use our results. However, if we would have would have used incorrect time-dynamics (e.g. giving input at incorrect times, or stopping the signal too late), then we can NOT use the results.

5.2 Prediction uncertainty

The reason why we do predictions is that we want to test our model, to test if we have to reject the model. If we have an underestimation of the model uncertainty, we might reject the true model.

5.3 Making a new measurement

We have three possible outcomes:

1. None of the models are able to explain the new data (e.g. if the measurment value is at 1.
2. The blue model is able to explain the data, the red model is not (e.g. at 5).
3. The red model is able to explain the data, the blue model is not (e.g. at 2).

In the case of 1), both of the model would be rejected. In 2) the red model is rejected. In 3) the blue model is rejected. Since the models are giving different behaviors at 6 seconds, we will (assuming the new data have the uncertainty given in the figure) be able to reject either or both of the models.