Updating Euler approximations to ODE systems

Hugh Murrell for Ben Murrell

May 16, 2020

Abstract

The order in which variables are updated matters.

1 Simulating the standard SIR model

The continuous SIR model [1]

$$\begin{aligned} \frac{dS(t)}{dt} &= -\beta\frac{S(t)I(t)}{N} \\ \frac{dI(t)}{dt} &= \beta\frac{S(t)I(t)}{N} - \gamma I(t)) \\ \frac{dR(t)}{dt} &= \gamma I_t \end{aligned} \tag{1}$$

can be discretised using a Euler approximation as

$$\begin{aligned} S_{t+\Delta t} &= S_t - \Delta t\beta\frac{S_tI_t}{N} \\ I_{t+\Delta t} &= I_t + \Delta t\beta\frac{S_tI_t}{N} - \Delta t\gamma I_t \\ R_{t+\Delta t} &= R_t + \Delta t\gamma I_t \end{aligned} \tag{2}$$

and this discrete system can be simulated correctly using the code in appendix A

2 Updating before sampling

Now consider the code in Appendix B where first `new_cases` are sampled and then `S` and `I` are updated and then `recoveries` are sampled and then `I` and `R` are updated. With this code we are simulating the following discrete system:

$$
\begin{aligned}
S_{t+\Delta t} &= S_t - \Delta t \beta \frac{S_t I_t}{N} \\
I_{t+\Delta t} &= I_t + \Delta t \beta \frac{S_t I_t}{N} - \Delta t \gamma (I_t + \Delta t \beta \frac{S_t I_t}{N}) \\
&= I_t + \Delta t \beta \frac{S_t I_t}{N} - \Delta t \gamma (I_{t+\Delta t} + \Delta t \gamma I_t) \qquad (3) \\
R_{t+\Delta t} &= R_t + \Delta t \gamma (I_t + \Delta t \beta \frac{S_t I_t}{N})
\end{aligned}
$$

3 Conclusions

The order in which sampling and updating occurs matters greatly

A

```
beta = R0 * gamma;
S = pop; I = init; R = 0;
C = I;
for i in 1:upper
    # do the sampling before the updates
    new_cases = min(rand(Poisson(beta*S*I/(S+I+R))),S)
    recoveries = min(rand(Poisson(gamma*I)),I)
    # now do the updates
    I += new_cases
    S -= new_cases
    I -= recoveries
    R += recoveries
    C += new_cases
    Rt = R0*(S/(S+I+R))
    # store the current counts in vectors
    NC[i],Sv[i],Iv[i],Rv[i],Cv[i],Rtv[i] = new_cases,S,I,R,C,Rt
end
```

B

```
beta = R0 * gamma;
S = pop; I = init; R = 0;
C = I;
for i in 1:upper
    # do some sampling and some updates
    new_cases = min(rand(Poisson(beta*S*I/(S+I+R))),S)
    I += new_cases
    S -= new_cases
    # now do more sampling and the rest of the updates
    recoveries = min(rand(Poisson(gamma*I)),I)
    I -= recoveries
    R += recoveries
    C += new_cases
    Rt = R0*(S/(S+I+R))
    # store the current counts in vectors
    NC[i],Sv[i],Iv[i],Rv[i],Cv[i],Rtv[i] = new_cases,S,I,R,C,Rt
end
```

References

[1] Wikipedia, *Compartmental models in epidemiology*, http://en.wikipedia.org/wiki/Compartmental_models_in_epidemiology .