CODE FOR IMPLEMENTING ABC-SMC

The code implements the ABC-SMC methods described in Guinat et al. (2018) Transboundary and Emerging Diseases 65(2), e264-e271.

ImplementABC_SMC.for: main program, loads data and implements ABC-SMC algorithm; requires two ASCII text files for inputs:

- MortalityData.txt - contains day (col. 1) and no. dead animals (col. 2) for each herd/flock.
- SMCInitFile.txt - a file with nine rows specifying:
    1. seed for random number generator (integer)
    2. initial number of animals in herd/flock (integer)
    3. number of observations in mortality data (i.e. the number of rows in MortalityData.txt) (integer)
    4. day of confirmation in herd/flock (i.e. day in mortality data when infection confirmed) (integer)
    5. initial number of infected animals (integer)
    6. flag indicating priors to use (integer; see below)
    7. number of particles to use (integer)
    8. number of SMC rounds to run (integer)
    9. initial acceptance tolerance to use (real)

Note: The input files are assumed to be in the same folder as the .exe file.

ComputePrior.for: computes the joint prior for the input parameters. Includes functions to compute PDFs needed for the priors.

multinomial.for: samples from a multinomial distribution.

gamdev.for: draws a sample from a Gamma(G,H) distribution with mean G/H and variance G/H^2.

RealSort.for: sorts a (real) vector of numbers into ascending order.

SimulateOutbreak.for: generates a simulated outbreak of ASF in a herd/flock for given input parameters.

UpdatePopulations.for: updates the populations at each time-step in the model.

howmany.for: determines the number of transitions of each type in a time-step.

binomial.for: samples from a Binomial(n,p) distribution, where n is the population size and p is the probability.

**poisson.for:** samples from a Poisson(mu) distribution, where mu is the distribution mean.

normal.for: samples from a Normal(0,1) distribution.

rangen.for: simulates a Uniform(0,1) random number.

gammaln.for: computes the log gamma function.

Note: The code for the model was developed and built using Microsoft Visual Studio 2010 and Intel Visual Fortran Composer XE 2013. It was implemented for a 64-bit architecture. All the source code needed to run the model is contained here (i.e. there are no external libraries/routines required), but the code has not been tested with other compilers and so is not guaranteed to work with them.

CODE STRUCTURE AND DEPENDENCIES
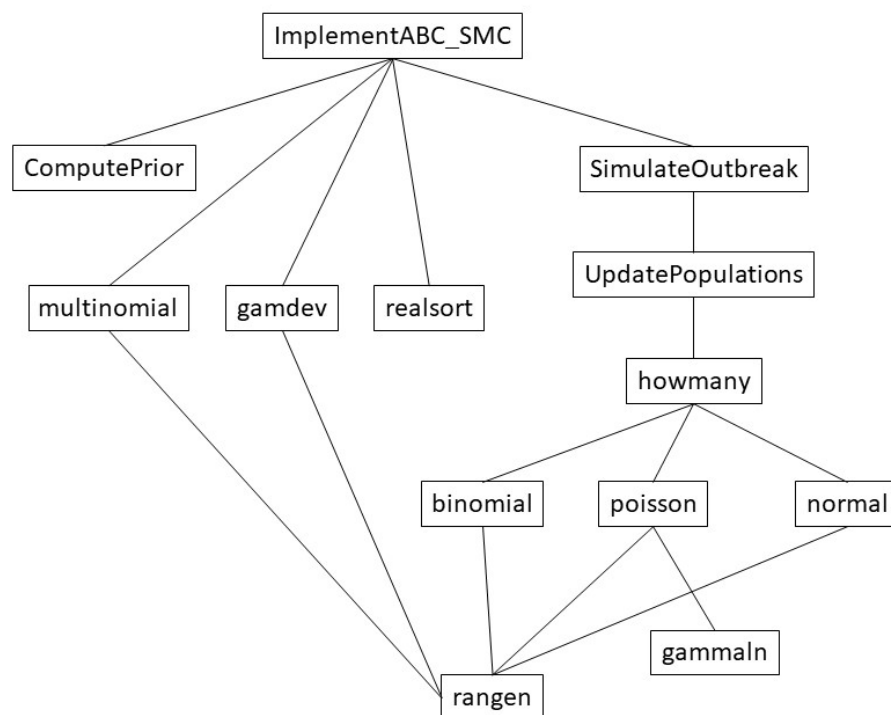
RUNNING THE CODE

The ABC-SMC scheme is run using the file **ImplementSMC.exe** generated by compiling and linking the code.

Inputs: The two ASCII files, one containing the data (MortalityData.txt) and one which sets up the scheme (SMCInitFile.txt) (see above for details), need to be in the same folder as the .exe file.

Outputs: In this folder, there also needs to be a subfolder called "Outputs", where the code writes the outputs. There are five output files:

- tolerance.txt - round number (col. 1) and acceptance tolerance for the round (col. 2)
- parsamp.txt - round number (col. 1), particle number (col. 2) and parameter values for the particle (cols 3-9)
- Mortality.txt - round number (col. 1), particle number (col. 2) and simulated mortality data for the particle (col. 3-end)
- weights.txt - round number (col. 1), particle number (col. 2) and weight for the particle (col. 3)
- theta.txt - round number (col. 1), particle number (col. 2) and parameter values for the particle (cols 3-9)

Parameters are (in order): (1) mean latent period; (2) shape for latent period; (3) mean infectious period; (4) shape for infectious period; (5) transmission parameter; (6) baseline mortality rate; and (7) time of introduction.

Note: the only difference between parsamp.txt and theta.txt is parsamp.txt is written to once a particle is accepted, while theta.txt is written to once a round has been completed. Accordingly, it is best to use theta.txt to monitor convergence of the scheme while it is still running.

PRIORS

Flags for priors are:

1. informative for all parameters
2. informative for all parameters, except transmission rate
3. informative for all parameters, except transmission rate and baseline mortality rate
4. non-informative for all parameters, except for transmission rate
5. non-informative for all parameters, except for baseline mortality rate
6. non-informative for all parameters

Priors are (currently):

parameter	informative prior	non-informative prior
mean latent period	Gamma(mean=1,shape=2)	Uniform(0,5)
shape for latent period	Gamma(mean=1,shape=1)	Uniform(0,5)
mean infectious period	Gamma(mean=5,shape=20)	Uniform(0,10)
shape for infectious period	Gamma(mean=5,shape=20)	Uniform(0,10)
transmission parameter	Gamma(mean=1,shape=2)	Uniform(0,5)
baseline mortality rate	Gamma(mean=0.0002,shape=1)	Uniform(0,0.005)

The prior for the time of introduction is Uniform(T1,T2), where:

$T1$=first day with mortality data-30

$T2$=day of confirmation.