

GUIDE TO THE CODE

CALIBRATION AND EVALUATION OF INDIVIDUAL-BASED MODELS USING APPROXIMATE BAYESIAN COMPUTATION

Elske van der Vaart, Mark Beaumont, Alice Johnston & Richard Sibly
Ecological Modelling (2015)

This guide offers a brief introduction to the code that accompanies the above paper. All code can be found a **figshare** repository: <http://dx.doi.org/10.6084/m9.figshare.1412720>, which is presumably where this guide was obtained in the first place. If not, check there for possible updates! In the following, familiarity with the paper itself is assumed.

There are two kinds of code in the repository: NetLogo code (.nlogo) and *R* code (.R). Both NetLogo and *R* are programming environments that are freely available for all operating systems; NetLogo is for IBMs, while *R* is for statistics. Accordingly, the paper's earthworm IBM is written in NetLogo, while its ABC calculations, tables and plots come from *R*.

This guide will explain how to look at the earthworm IBMs, and how to re-create all of the ABC results given in the paper. Any questions, e-mail <elskevdv@gmail.com>.

- Elske van der Vaart, 19 May 2015

LOOKING AT THE EARTHWORM IBMS

As stated, the earthworm IBM is written in NetLogo, which can be downloaded from here: <http://ccl.northwestern.edu/netlogo/>

Working with NetLogo models is very easy; NetLogo itself can be installed with a graphical installer, and then individual models can be opened from a drop-down menu. Each model has a visual interface and very human-readable code that can be accessed from another tab; for the earthworm IBM, the visual interface already offers some parameters to play with and some graphs to assess the model's performance as it's running.

In the files downloaded from the **figshare** repository, there's actually 8 different .nlogo files: The title of each file refers to the empirical data it is attempting to replicate, with a subscript 0 or 1 that indicates whether it's the "simple" or "full" version of the model as described in the paper, respectively. The "full" version of the model was originally developed by Alice Johnston [1], while the "simple" version was developed specifically for this paper.

The file titles relate to the empirical data and the results presented in the paper as follows:

- g2002 – [2] – Panel A in Figure 4;
- ge2003 – [3]– Panel D in Figure 4;
- rv1990v – [4], variable condition – Panels B & E in Figure 4;
- rv1990c – [4], constant condition – Panels C & F in Figure 4.

To look at any of these models, simply open them from NetLogo's drop-down menu. Press "Setup" and then "Go \rightleftharpoons " in the 'Interface' tab to watch them run. The graph will display the model's output in grey and the empirical data in black; the standard parameter settings are the "literature values" as described in the paper, but these can be changed using the input boxes. The model code itself is under the 'Code' tab.

RUNNING THE ABC ANALYSES IN R

To reproduce the paper's ABC results, first download R: <http://www.r-project.org>. R can be installed with a graphical installer, and then from within R, it's necessary to download a number of R **packages**: *car* and *Hmisc* in the first instance. This should be possible from a drop-down menu, but as the process differs across versions and operating systems, I'm refraining from giving specific instructions. Make sure to download all dependencies as well.

the simulation results

It will also be necessary to download all of our simulation results, available from a separate **figshare** repository due to licensing issues: <http://dx.doi.org/10.6084/m9.figshare.1412720>. The files in this repository are the following:

- *all_data.txt* – a file with all of the empirical data as plotted in Figure 4;
- *priors_full.rds*, *results_full_1.rds* and *results_full_2.rds* – all one million priors and results for the “full model”, stored in R's file format; each row in *priors* contains the parameter settings for the corresponding row in *results*, split over two files to stay under figshare's 250 mB limit;
- *priors_full_1e5.txt* and *results_full_1e5.txt* – the first one hundred thousand rows of the files described above, provided for those who might have trouble with a one million row file, or who would like to be able to look at the priors and results outside of R;
- *priors_simple.rds*, *results_simple_1.rds*, *results_simple_2.rds*, *priors_simple_1e5.txt* and *results_simple_1e5.txt* – the same as above, but for the “simple model” as used for the model selection example;
- *cross_est.rds*, *cov_est.rds* and *cross_sel.rds* – the cross-validation and coverage results presented in the paper, for parameter estimation and model selection respectively; these take quite long to produce and are therefore included in their entirety (along with the code to replicate them).

organising all files correctly

To actually run the code, open the file ‘Commands.R’ in R (this can again be done from a drop-down menu). In theory, setting two paths in this file and then executing it (for instance, by copying and pasting it into R's console) will produce all of the results figures and tables in the paper. However, it does assume a certain file structure, which looks like this:

- Main folder: name of your choice, e.g., ‘worms’
 - Subfolder: ‘results’ [all files from the ‘Runs and Results’ figshare]
 - *all_data.txt*
 - *priors_full.rds*
 - *results_full_1.rds*
 - *results_full_2.rds*
 - *priors_full_1e5.txt*
 - *results_full_1e5.txt*
 - *priors_simple.rds*
 - *results_simple_1.rds*
 - *results_simple_2.rds*
 - *priors_simple_1e5.txt*
 - *results_simple_1e5.txt*

- Subfolder: 'src'
 - Subfolder: 'models' [all .nlogo files from the 'Code' figshare]
 - g2002_0.nlogo
 - g2002_1.nlogo
 - ge2003_0.nlogo
 - ge2003_1.nlogo
 - rv1990c_0.nlogo
 - rv1990c_1.nlogo
 - rv1990v_0.nlogo
 - rv1990v_1.nlogo
 - Subfolder: 'R' [all .R files from the 'Code' figshare]
 - ABCObject.R
 - Commands.R
 - Coverage.R
 - CrossValidation.R
 - ModelSelection.R
 - ParameterEstimation.R
 - PosteriorChecking.R

actually running the code – Commands.R

The code in 'Commands.R' is the 'skeleton' for what's in the paper: It executes all of the necessary commands, but most of the 'real work' occurs in other scripts (i.e., all of the other *R* scripts in the 'R' folder described above). These scripts have sensible names and are organised into functions, with some annotation; please look at them to truly see what's happening. The only slightly opaque script is "ABCObject.R", which contains functions to make sure objects of class "abcObject" are printed to the console in a pretty fashion.

Some of these functions use code from the very, very useful *R* package *abc* [5]; the reason we don't use this package directly is because it normalises simulation results by a metric that's unsuitable in our case, as described in the paper.

notes about Commands.R up until 'Doing Posterior Checking'

Now, to run the 'Commands.R' script up until the 'Doing Posterior Checking' section, set the 'f.path' variable to the location of the 'main folder' (e.g., 'worms') as described above. Everything should just run, but there are two things I'd like to note:

Firstly, 'Commands.R' reads in the full one million rows of priors and results. In my experience, this proves impossible on a machine with less than 16 GB of RAM. In that case, read in the '_1e5.txt' variants of these files instead, by switching the relevant 'TRUE' and 'FALSE' blocks in 'Commands.R'; things will come out slightly different from the paper, but the overall impression should be the same.

Secondly, most of this script is re-creating results from scratch, using just the simulation results as input. However, cross-validation and coverage take a long time to run, so the script just performs a few examples of each, and then reads in the results of the calculations I've done. Thus, these figures should look *exactly* like those in the paper (provided the full one million runs are used, of course!).

notes about 'Doing Posterior Checking'

The posterior check as performed in the paper – Figure 4 – has its own section in 'Commands.R' because it is a bit trickier to run. It will require downloading NetLogo (as described under 'Looking at the Earthworm IBMs') and the R package *RNetLogo* [6, 7], as well as setting the 'nl.path' variable to the correct location. In theory this should be easy – *RNetLogo* can be downloaded from within R just like any other package – but in practice it may prove difficult. *RNetLogo* relies on the R package *rJava* which in turn requires a Java installation; that's a lot of opportunities for things to go wrong.

One problem we have encountered is when R and Java aren't installed using the same (32 or 64) bit version; then R throws an error which can be fixed by re-installing the correct Java version. The *RNetLogo* package itself is very well documented and its manual, vignette & associated papers [6, 7] should solve many problems. (It's a lovely package – I recommend it strongly to anyone working with both R and NetLogo for whatever reason!) However, feel free to contact me if this step (or indeed, any step) proves difficult.

REFERENCES

1. Johnston, A.S.A., et al., *An energy budget agent-based model of earthworm populations and its application to study the effects of pesticides*. Ecological Modelling, 2014. **280**: p. 5-17.
2. Gunadi, B., C. Blount, and C.A. Edwards, *The growth and fecundity of Eisenia fetida (Savigny) in cattle solids pre-composted for different periods*. Pedobiologia, 2002. **46**: p. 15-23.
3. Gunadi, B. and C.A. Edwards, *The effects of multiple applications of different organic wastes on the growth, fecundity and survival of Eisenia fetida (Savigny) (Lumbricidae)*. Pedobiologia, 2003. **47**: p. 321-329.
4. Reinecke, A.J. and S.A. Viljoen, *The influence of feeding patterns on growth and reproduction of the vermicomposting earthworm Eisenia fetida (Oligochaeta)*. Biology and Fertility of Soils, 1990. **10**: p. 184-187.
5. Csillery, K., O. François, and M.G.B. Blum, *abc: An R package for approximate Bayesian computation (ABC)*. Methods in Ecology and Evolution, 2012. **3**: p. 475-479.
6. Thiele, J.C., W. Kurth, and V. Grimm, *RNetLogo: An R package for running and exploring individual-based models implemented in NetLogo*. Methods in Ecology and Evolution, 2012. **3**: p. 480-483.
7. Thiele, J.C., W. Kurth, and V. Grimm, *Facilitating parameter estimation and sensitivity analysis of agent-based models: A cookbook using NetLogo and R*. Journal of Artificial Societies and Social Simulation, 2014. **17**: p. 11.