



BICI V1.0

Bayesian Individual-based Compartmental Inference

Christopher M. Pooley^{1,2}, Andrea B. Doeschl-Wilson¹, and Glenn Marion²

¹ The Roslin Institute, The University of Edinburgh, Midlothian, EH25 9RG, UK.

² Biomathematics and Statistics Scotland, James Clerk Maxwell Building, The King's Buildings, Peter Guthrie Tait Road, Edinburgh, EH9 3FD, UK.

Table of Contents

1 Introduction	4
1.1 Downloading	4
1.2 Getting started	4
1.3 Description	5
2 The Model	7
2.1 Defining compartments and transitions	7
2.2 Equations	8
2.3 The ‘Age’ classification	9
2.4 The ‘Time’ classification	10
2.5 Derived quantities	10
2.6 Distributions	10
2.7 Completion	11
3 Simulation	11
3.1 Parameters	11
3.2 Initial population	12
3.3 Running	12
3.3 Output	14
3.3.1 Populations	14
3.3.2 Individuals	16
3.3.3 Statistics	16
3.3.4 Generating data	16
4 Inference	18
4.1 Data	18
4.1.1 State data	20
4.1.2 Presence data	20
4.1.3 Transition data	22
4.1.4 Move data	22
4.1.5 Capture data	22
4.1.6 Capture ID data	24
4.1.7 Capture PD data	24
4.1.8 Population data	24
4.1.9 Derived data	24
4.1.10 Individuals	25
4.2 Prior	26
4.3 Posterior	27
4.3.1 Starting inference	27

4.3.2 Trace plots.....	29
4.3.3 Probability distributions.....	30
4.3.4 Correlation	30
4.3.5 Dependency	30
4.3.6 Transition distributions.....	34
4.3.7 Individual timelines	34
4.3.8 Statistics	35
4.3.9 Population plots	37
4.3.10 Posterior predictive population plots	39
4.3.11 Derived.....	39
5 Input and output.....	39
5.1 Loading and saving	39
5.2 Exporting	40
5.3 Importing.....	40
6 Examples.....	41
6.1 SIR model.....	41
6.2 The SEIR model.....	43
6.3 Disease transmission experiments	44
6.4 Population with births and deaths	46
6.5 SIS model with births and deaths.....	46
6.6 Predator-prey model	47
6.7 Spatial diffusion model.....	48
6.8 Spatial disease model	48
6.9 Location-based spatial disease model.....	49
6.10 Spatial disease model with births and deaths.....	50
7 Code.....	50
8 License and warranty.....	51
9 Citing BICI.....	51
10 Plans for BICI v2.0	51
Acknowledgments	51
References	51

1 Introduction

Compartmental models have long been used as a means of understanding the collective dynamics of interacting agents, with notable applications in epidemiology and ecology. BICI allows for arbitrary compartmental model specification and performs simulation and/or inference on that model.

For inference BICI takes in a variety of individual-based or population-based data, priors can be specified from a large range of possibilities, and the outputs consist of posterior trace plots for model parameters, distributions, correlations, visualisations of transitions, dynamic population estimates, and summary statistics (means and 95% credible intervals) as well as diagnostics.

A detailed description of the statistical model underlying BICI, along with the Bayesian inference methodology, is given in an accompanying [paper](#) [1]. The focus of this manual is on the practicalities of analysing real world data and interpreting the results.

1.1 Downloading

BICI is freely available under the GNU General Public License, and can be downloaded from www.mkodb.roslin.ed.ac.uk/EAT/BICI.html.

Depending on your platform, the following instructions explain how BICI can be run:

- **Windows** – Download the file “BICI_v1.0_windows.zip” and unzip. BICI is run by clicking on the “BICI.exe” icon.
- **Linux** – Download the file “BICI_v1.0_linux.tar.gz”. This can be extracted by using the terminal command “tar -zxvf BICI_v1.0_linux.tar.gz”. The code is executed using “./BICI”.
- **Macintosh** – Download the file “BICI_v1.0_Mac.zip”. BICI is run by clicking on the “BICI.app” icon (if the error message “BICI can’t be opened because it is from an unidentified developer...” appears, right click on “BICI.app” and select “Open” to give the option to run).

1.2 Getting started

Figure 1 shows the page displayed when BICI is first loaded. The main menu on the left (Fig. 1A) is used to navigate arbitrarily from page to page. To start using BICI three options are available: a previous analysis can be loaded (Fig. 1B, note BICI uses a special ‘.bici’ file format for loading and saving analyses, as described in §5.1), a new analysis can be started (Fig. 1C), or one of the example applications can be investigated (Fig. 1D).

The examples demonstrate a number of simple models and data scenarios that illustrate the capabilities of BICI in a variety of different situations (a comprehensive description of these examples is given in §6). New users are encouraged to try these first and spend some minutes exploring the software to get a feel for how BICI’s interface works.

BICI

- > Home
- > Description
- > Model
- > Simulation
- > Inference

(A)

(B)

(C)

(D)

SIR model [?]

SEIR model [?]

A description of how to use this software is provided in the attached manual. Further details are given in a paper hosted on bioRxiv.

PDF Manual
Paper

Examples [?]

- > EX 1: Complete knowledge of events
- > EX 2: Known initial state and recoveries
- > EX 3: Known recoveries only
- > EX 4: Periodic disease status data
- > EX 5: Diagnostic test results
- > EX 6: Future and past prediction
- > EX 7: Time classification
- > EX 8: Non-Markovian latent period
- > EX 9: Disease diagnostic tests
- > EX 10: Environmental and diagnostic tests

© BICI 2019

Figure 1 – The home page. A: Main menu, B: Load previous analysis, C: Start a new analysis, D: Examples, E: Click on [?] symbols to provide more information.

As an illustration, example EX 1 can be selected (Fig. 1D). Pressing the “Start” button on the ‘Simulation→Run’ page (here we use the notation “ \rightarrow ” to mean “submenu”) leads to simulated output from the model, and pressing “Start” on the ‘Inference→Posterior→Run’ page leads to inference being carried out (based on previously loaded transition data in the case of EX 1) and several different visualisations of the posterior.

Note, the examples can be modified in any way, *e.g.* by making changes to the model/prior/data, but their default settings are restored when reloaded from the home page (Fig. 1D). Whilst exploring the software make use of the many [?] help buttons (*e.g.* Fig. 1E) that provide much of the information outlined in this manual.

This document broadly follows the order of the items on the main menu (Fig. 1A), starting with analysis description below and model definition (§2) and going on to describe simulation (§3) and inference (§4). Section 5 then discusses input and output (*e.g.* importing model/data/prior and exporting graphs) and, finally, details of the examples are given in §6.

1.3 Description

As shown in Fig. 2A, BICI allows users to provide a brief description of their model, data, and analysis. This is not only useful to keep track for personal use, but also makes life easier for others to understand what has been done and why. The description can be edited by clicking on the “Edit” button (Fig. 2B). Note, bullet points are automatically generated for each carriage return in the editable text box. Once complete, click the “Next” in the bottom right-hand corner of the screen (Fig. 2C). It should be pointed out that the “Next” buttons on each page are just for convenience. In fact the menu on the left-hand side (Fig. 1A) can be used to navigate to any page without loss of information.

BICI

- > Home
- > Description
- > Model
- > Simulation
- > Inference

Description [?]

A • This simple epidemiological model describes how infection spreads within a population.

- The model assumes that individuals can be classified as susceptible "S", infected "I" or recovered "R".
- Transitions in state occur when individuals become infected or recover (the rate at which these occur is characterised by the model parameters β and γ).
- The data for inference consists of the infection and recovery times of individuals (generated by simulating from the model), which represents complete knowledge of the system.
- Running inference and viewing the posterior probability distributions shows that BICI is able to accurately estimate the parameter values used to generate the data.
- The basic reproductive ratio R_0 is defined as the number of infections caused by an infected individual in an otherwise susceptible population. R_0 can be derived from the model parameters (see the 'Derived' tab in the model section) and posterior distributions are shown when inference is performed.

(B) (C)

Edit

Next »

© BICI 2019

Figure 2 – Description. A: Text panel giving a description of the data, model, and analysis (here for EX 1); B: Edit this description, C: Go to the next page.

BICI

- > Home
- > Description
- < Model
 - > DS
 - + Class. [?] **(A)**
 - > Age [?]
 - > Time [?]
 - > Derived
 - > Distributions
- > Simulation
- > Inference

This shows the classification of individuals by disease status (DS). The notation $\{\}$ indicates the total population in the 'I' compartment.

Export Import [?]

```

graph LR
    S[S] -- "β {I}" --> I[I]
    I -- "γ" --> R[R]
  
```

(B) + Compartment [?] (C) + Transition [?] (D) (E) Done!

© BICI 2019

Figure 3 – Compartmental Model. This shows the SIR compartmental model from EX 1 (the compartments represent 'S' susceptible, 'I' infected and 'R' recovered from disease). A: Add a classification, B: Add a compartment, C: Add a transition, D: Zoom in and out of the model, E: Complete definition of the model.

2 The Model

This section describes different aspects of the model definition.

2.1 Defining compartments and transitions

The compartmental model is defined in the following three ways:

Classifications – Individuals are defined by one or more discrete quantities of interest (e.g. disease status, sex, location) and each of these is incorporated into the model as a so-called “classification”. New classifications are added into the model by pressing on the “ Class.” button (Fig. 3A). The example in Fig. 3 shows the case of a single classification called ‘DS’, standing for “disease status”.

Compartments – For each classification, individuals can take one or more different states or “compartments”. The example in Fig. 3 contains three compartments that define the possible disease states an individual can take: ‘S’ represents susceptible to disease, ‘I’ represents infected, and ‘R’ represents recovered/removed/dead. Collectively this is known as the “SIR model”. Compartments are added to the model by clicking on the “ Compartment” button (Fig. 3B) and then placing them onto the screen. Clicking on an existing compartment allows its name to be edited, its colour to be change or its removal.

Transitions – The movement of an individual from one compartment to another is referred to as a “transition”. Figure 3 contains two transitions represented by black arrows: one from ‘S’ to ‘I’, which represents infections, and one from ‘I’ to ‘R’, which represents recoveries. A transition is added to the model by clicking on the “ Transition” button (Fig. 3C). It is necessary to choose from five different types:

- *Markovian*: For individuals in the initial compartment the transition to the final compartment happens with a certain probability per unit time, or “rate”. Precisely how this rate is defined will be explained in the next section (alternatively if the ‘Time’ option is selected, the reciprocal of the rate, or the average duration, is defined instead). Markovian transitions occur independent of when an individual entered the initial compartment. Such an approach is valid for random processes, for example the random contacts between different members of a population leading the spread of infection.
- *Gamma*: This assumes a gamma probability distribution for the duration an individual remains in the initial compartment before making the transition to the final compartment. This requires both a mean duration and shape parameter to be defined¹. An example application of the gamma distribution is to model the incubation period (time duration between becoming infected and then becoming infectious) as used in the SEIR model in EX 8. In reality such a transition is expected to be non-Markovian because the time an individual becomes infectious is related to when they were infected as a result of complex processes underlying immune system dynamics.
- *Weibull*: This assumes a Weibull probability distribution for the duration an individual remains in the initial compartment before making the transition to the final compartment. This requires both a scale parameter and shape parameter to be defined. The Weibull

¹ A high shape parameter k corresponds to very little deviation about the mean, whereas low k corresponds to substantial variance ($k=1$ implies a Markovian exponential distribution).

distribution is somewhat similar to the gamma distribution, but is used in preference under certain circumstances.

- *Source*: This allows the rate at which individuals enter the system to be defined². These new individuals may come from some external reservoir or represent births (as is commonly used in ecological applications, e.g. EX 19). Note, unlike the transition types above, this only requires specification of the compartmental state into which individuals enter.
- *Sink*: This allows the rate at which individuals leave the system to be defined. In the ecological setting these transitions can be used to represent the deaths of individuals (e.g. EX 19).

Once the transition type has been selected it can be added to the model by clicking on the relevant compartment(s) (intermediary points can also be added to map out a path instead of just a line, e.g. this could be used to add a non-straight arrow going from ‘R’ back to ‘S’ again in Fig. 3 to incorporate waning immunity).

2.2 Equations

Editable equations provide the way in which various quantities within the model are defined. Most obviously they are used to identify key expressions for the transitions above, but they are also used in other contexts (for example the observation model), as described later.

Clicking on the transitions in the model (Fig. 3) allows them to be edited. Figure 4A shows the equation used to define the rate at which susceptible individuals become infected. Parameters can be added to this expression by clicking on Fig. 4B or typing them directly into the editable text area. Note, parameters are surrounded by square brackets, such as $[\beta]$. If the model parameter used depends on the status of the individual in another classification the “_” symbol is used followed by this dependency. For example, suppose there exists a second classification ‘Sex’ that contains compartments ‘M’ for male and ‘F’ for female. Entering $[\beta_{_Sex}]$ results in two model parameters β_M and β_F which are separately used to calculate the transition rates for the male and female individuals, respectively. Dependency on more than one classification can also be added (e.g. $[\beta_{_Sex,Vac}]$ depends on both sex and vaccination status).

A so-called “population” can be added to the equation by clicking on Fig. 4C (or alternatively by typing the required compartments surrounded by curly brackets). The example in Fig. 4A $\{I\}$ represents the total population in the ‘I’ infected compartment. Compartments from different classification can be added to the expression, e.g. $\{I,M\}$ would give the number of infected males. If the name of the classification itself is used, this is replaced by the compartment of the individual undergoing the transition, e.g. supposing ‘Group’ was a classification for different groups of

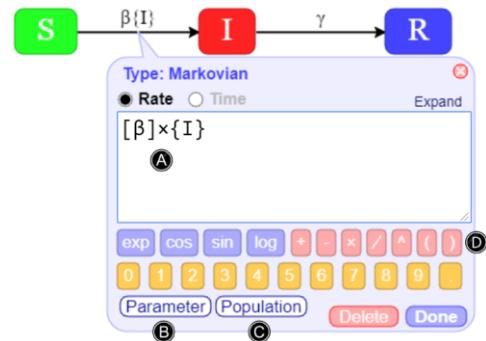


Figure 4 –Equations. A: Text area for equation definition, B: Add a parameter, C: Add a population, D: Other quantities.

² In cases in which multiple classifications are defined, sources and sinks can only be defined in a single classification.

individuals, '{I,Group}' would give the total number of infected individuals for the group in which a given individual resides.

Figure 4D shows other quantities that can be added to the equation. Operators: '+' for adding, '-' for minus, 'x' or '*' for multiplication, '/' for divide, and '^' for power. Rounded brackets can be used to determine the order in which operations are performed (the standard order of precedence '^', 'x', '/', '+', '-' is used by default). Furthermore, the exponential 'Exp(...)', logarithmic 'Log(...)', sine 'Sin(...)', and cosine 'Cos(...)' functions can be added, along with integer and floating point numbers.

When considering transitions, BICI also provides a general way of differentiating the expression used for an individual depending on the state of that individual. This is achieved by writing multiple lines with a filter applied to each line. For example suppose we consider a model with susceptible 'S' and infected 'I' individuals which are male 'M' or female 'F'. The following expression can be used to define the rate at which individuals die:

$$\begin{aligned} S &:[\mu] \\ I,M &:[m] \times [\mu] \\ I,F &:[q] \times [\mu] \end{aligned} \quad (1)$$

Note, this is made up of separate lines in the editable text box in Fig. 4A, with each line consisting of a filter followed by a colon and then the equation expression. Consequently, Eq.(1) means that susceptible individuals die with rate parameter μ , infected males die at a higher infection-induced mortality rate determined by factor m , and infected females have an equivalent factor q . Examples of such filtering can be seen in EX 27, which looks at disease induced mortality, and EX 34 and 35, which model the spatial spread of disease between farms.

As well as the normal classifications that can be added to the model to distinguish individuals based on their properties (those above the “⊕ Class.” button in Fig. 3A), BICI incorporate two further classifications related to individual age and the global timeline.

2.3 The 'Age' classification

This special classification can be viewed by clicking on the 'Model→Age' tab (Fig. 3B). An example is shown in Fig. 5, which contains three age compartments. Once defined, other quantities in the model can depend on this classification, e.g. an age dependant mortality rate μ_{Age} as used in EX 22.

Age transitions are added to the model by clicking on the “⊕ Age transition” button (Fig. 5A) and entering the age at which the transition occurs. Transitions can be removed by clicking on the link (Fig. 5B) and selecting “Delete”.

When a source is defined in the model, by default individuals entering the system have an equal probability of entering any of the age compartments. However, in cases in which the source represents the birth of individuals, a very short initial age range is added via which individuals enter



Figure 5 –Age Classification. This shows an example in which the age of individuals is divided into the compartments 'Ao-1' for individuals up to 1, 'A1-2' for those between 1 and 2 and 'A2+' for those 2 and above. A: Add age transition, B: Click on transition to remove.

the system (e.g. in EX 22 the age range ‘A0-0.01’ is specified and the equation ‘A0-0.01:[v]{P}(1-[P]/[k])’ represents the source transition rate at which individuals are born).

If age dependant quantities are expected to have an underlying continuously varying profile (which is usually the case), a smoothing prior can be added when inference is performed (see §4.2).

2.4 The ‘Time’ classification

This special classification can be viewed by clicking on the ‘Model→Time’ tab (Fig. 3B). The global timeline can be split into discrete periods, which allows temporal variation in parameter values to be incorporated into the model. For example, two time periods might relate to before and after a disease intervention strategy (e.g. EX 7), or yearly intervals can account for variations in weather conditions. Figure 6 shows an example with 3 time compartments. Once defined, other quantities in the model can depend on this classification, e.g. the transition rate β_{Time} in EX 7.

Time transitions are added to the model by clicking on the “⊕ Time transition” button (Fig. 6A) and entering the time at which the transition occurs. Transitions can be removed by clicking on the link (Fig. 6B) and selecting “Delete”.

As above, if time dependant quantities are expected to have an underlying continuously varying profile, a smoothing prior can be added when inference is performed (see §4.2).

2.5 Derived quantities

These are quantities that are functionally dependant on the model parameters and/or populations (potentially through additional parameters), as defined by an editable equation (see §2.2). Derived quantities can be added to the model by going to the ‘Model→Derived’ tab (Fig. 3B) and clicking on the “⊕ Derived” button.

Derived quantities are used either when they are of physical relevance (e.g. in EX 1 the basic reproductive ratio R_0 is defined which characterises the rate with which an epidemic will proliferate) or when measured data relates to these derived quantities rather than directly to anything in the model itself (e.g. in EX 11 environmental pathogen level is assumed to be proportional to the number of infected individuals through an estimable factor).

2.6 Distributions

In some cases, rather than simply defining priors on parameters it can be appropriate to assume they are drawn from some specified distribution (which itself contains hyperparameters on which further priors are specified). Such distributions are added to the model by going to the ‘Model→Distributions’ page (Fig. 3B) and clicking on the “⊕ Distribution” button.



Figure 6–Time Classification. This shows an example in which the global time line is divided into the compartments: ‘T<40’ for times before 40, ‘T40-70’ for times between 40 and 70 and ‘T>70’ for times after 70.

A: Add time transition, B: Remove transition.

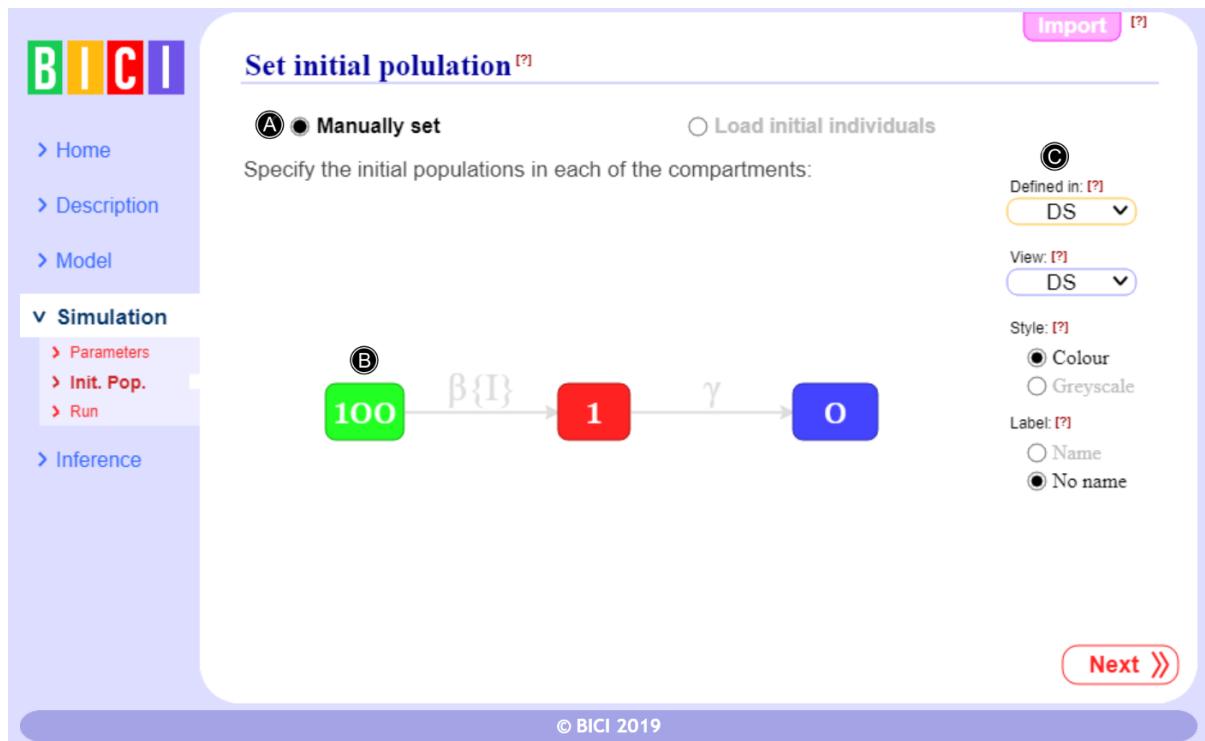


Figure 7 – Initial population. This page is used to define the initial population before simulation is carried out. A: Select whether manually defined or loaded from a file, B: Define population sizes in compartments, C: Various options: ‘Defined in’ shows classification in which population sizes are defined, ‘View’ shows classification being viewed, and ‘Style’ and ‘Label’ define how compartments are displayed.

When this specified distribution is normal with zero mean and standard deviation σ , the dependent parameters are referred to as “random” effects. An example is given in EX 16, in which a normally distribution “group effect” is added to capture the fact than epidemics proceed faster or slower in different epidemic groups due to differences in environmental conditions. Here the estimable parameter σ provides a measure of this environmental variation.

2.7 Completion

Once specification of the model is complete, the user clicks the “Done!” button in Fig. 3E (if subsequent changes need to be made to the model the “Edit” button is selected). This leads to the user being given the choice between simulation and inference, which are the subject of the next two sections.

3 Simulation

Simulation relies on a specification of model parameters along with the initial state and stochastically predicts the dynamics of the compartmental model over a specified time period.

3.1 Parameters

Before a simulation can be started it is necessary to first specify all model parameters on the ‘Simualtion→Parameters’ page. When a model is first setup parameters are set to the value “Unspecified”, written in green. Clicking on these opens up an editable text box into which a numeric value is entered (note, generally speaking editable quantities in BICI are written using green text).

In cases in which parameters have dependencies, these can be clicked on to reveal the individual elements (similarly, clicking on the individual elements combines them together again). For example, EX 22 contains the transmission rate parameter β_{Time} . If this is set to some value, say $\beta_{\text{Time}}=0.1$, then during simulation β_{Time} will be constant (and so independent of time). However, clicking on the ‘Time’ subscript reveals the parameters $\beta_{T<40}$ and $\beta_{T>40}$, which can then be separately set to define β differently before and after time $t=40$.

3.2 Initial population

The composition of the initial population is defined on the ‘Simualtion→Init. Pop.’ page, as shown in Fig. 7. Two options must be selected from (Fig. 7A):

Manual – This allows the user to set the initial population sizes for different compartments in the model (in the example in Fig. 7B the initial population has 100 susceptible individuals and a single infected). In cases in which the model has more than one classification, the one used to specify population sizes is selected from the “Defined in” drop-down menu (Fig. 7C). For other classifications (selected using the “View” drop-down menu) percentages are set that define the probability of individuals being in the various compartmental states (the individual states themselves are randomly sampled based on these probabilities when simulation is actually started).

Load initial individuals – This option requires the user to upload a text file providing information about the initial population. This file must consist of lines that each contain an individual ID followed by a tab separator, and then the initial compartment (multiple classifications are tab separated). An example of this is given in Fig. 8, which is appropriate for a model with two classifications: ‘DS’ which contains compartments ‘S’ and ‘I’ and ‘Sex’ which contains compartments ‘Male’ and ‘Female’.

Ind. 1	S	Male
Ind. 2	I	Female
Ind. 3	I	Female
Ind. 4	S	Female
Ind. 5	I	Male
:	:	:

Figure 8 – Load initial individuals. An example of an input file. Columns are tab-separated, with the first giving individual IDs and others giving compartmental states.

3.3 Running

Simulations are started on the ‘Simulation→Run’ page. The time range over which they are performed must first be specified. Clicking on “advanced options” reveals two other options:

- **Individual limit** – The maximum number of individuals can be set (by default this is 10^5). Such a limit is placed to avoid BICI crashing due to excessive memory usage (this typically only becomes a problem when the model specification leads to the number of individuals diverging³ during a simulation).
- **Number of simulations** – Usually this is just one, but multiple simulations can also be performed (this can be used to assess stochastic variation over a large number of simulations or the resulting ‘samples’ may be exported from BICI for use elsewhere).

To begin a simulation click on the “Start” button. Typically it takes just a few seconds or less. We now discuss the various ways in which the results can be viewed.

³ I.e. approaching infinity.

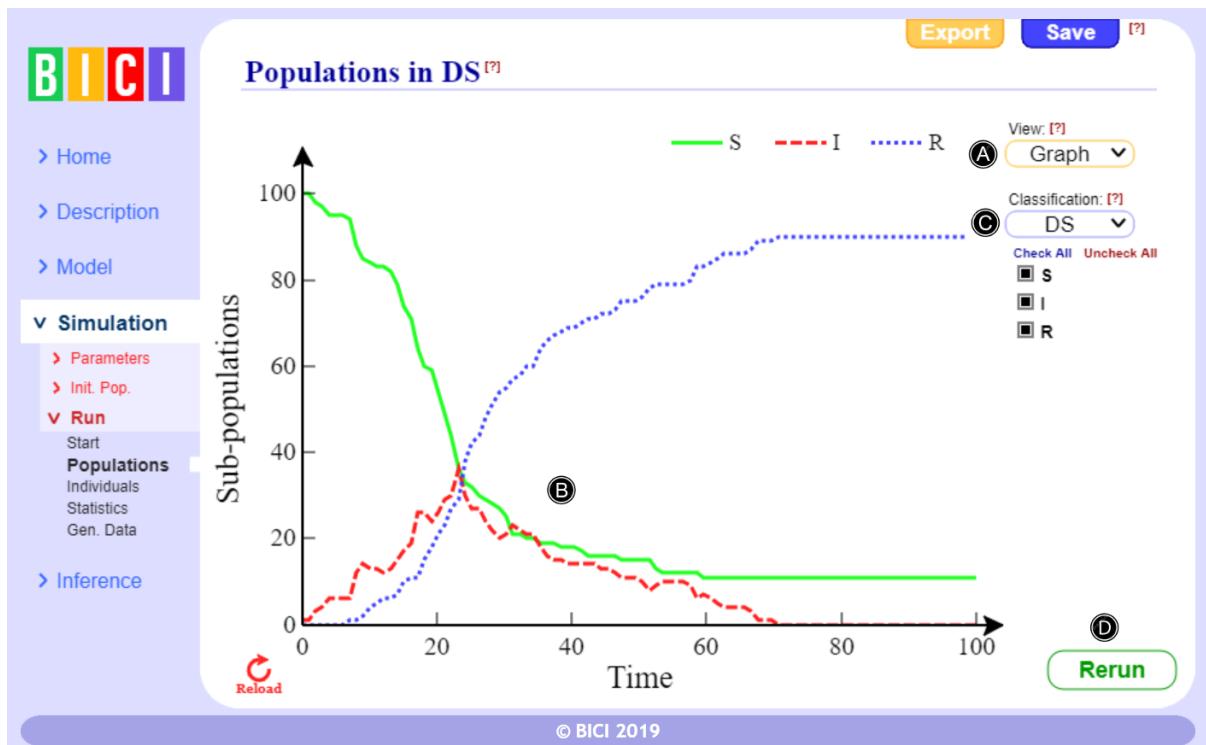


Figure 9 – Population: graph view. Simulated population dynamics for the SIR model (EX 1). A: Select the type of view, B: Graph showing population sizes as a function of time, C: Choose which populations to show, D: Rerun simulation.

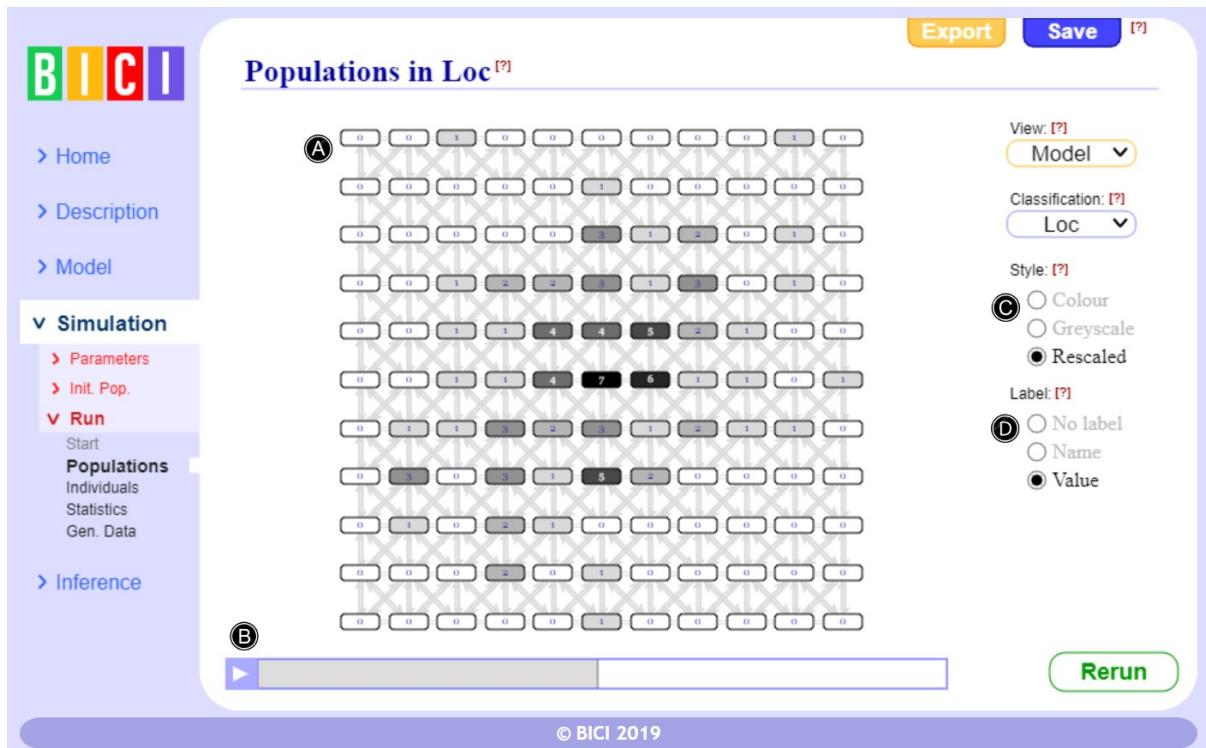


Figure 10 – Population: model view. A snapshot of simulated population dynamics for a spatial model (EX 30). A: Shows populations in compartments (darker shading represents larger population size), B: Play button and time bar, C: Options for style, D: Options for label.

3.3 Output

3.3.1 Populations

The ‘Simulation→Populations’ page shows how populations dynamically change as a function of time. These variations can be viewed in two different ways (see options in Fig. 9A):

As a graph – Figure 9B shows an example taken from EX 1. Dynamic changes in the population sizes in the model compartments are plotted as a function of time (in cases in which multiple simulations are performed, shaded areas denote regions which contain 95% of the simulations). The observed classification can be changed (Fig. 9C) and individual compartments within this can be turned on or off by clicking on the relevant checkboxes. For models with more than one classification, further filters can be applied to focus on only those individuals in specified states (*e.g.* at a particular location).

As an animation – Figure 10A shows the spatial spread of individuals at a particular time point (taken from example EX 30). The model is shown in greyscale to represent population sizes in each of the compartments (with darker colours representing larger population sizes). Pressing the play button (Fig. 10B) starts an animation which shows the dynamic variation in these populations as a function of time. The animation can be paused, restarted or a particular time point can be viewed by clicking on the time bar. Several options can be selected regarding how the data is displayed: The colour scheme from the model or greyscale (Fig. 10C) can be used to represent population sizes (the ‘rescale’ option rescales the palette each frame to help highlight variation which would otherwise be difficult to see). The label on the compartments can be turned on or off and selected to include the compartmental name or not (Fig. 10D).

When multiple simulations have been performed, the “Simulation” drop-down menu can be used to select results for a particular simulation or for all of them simultaneously.

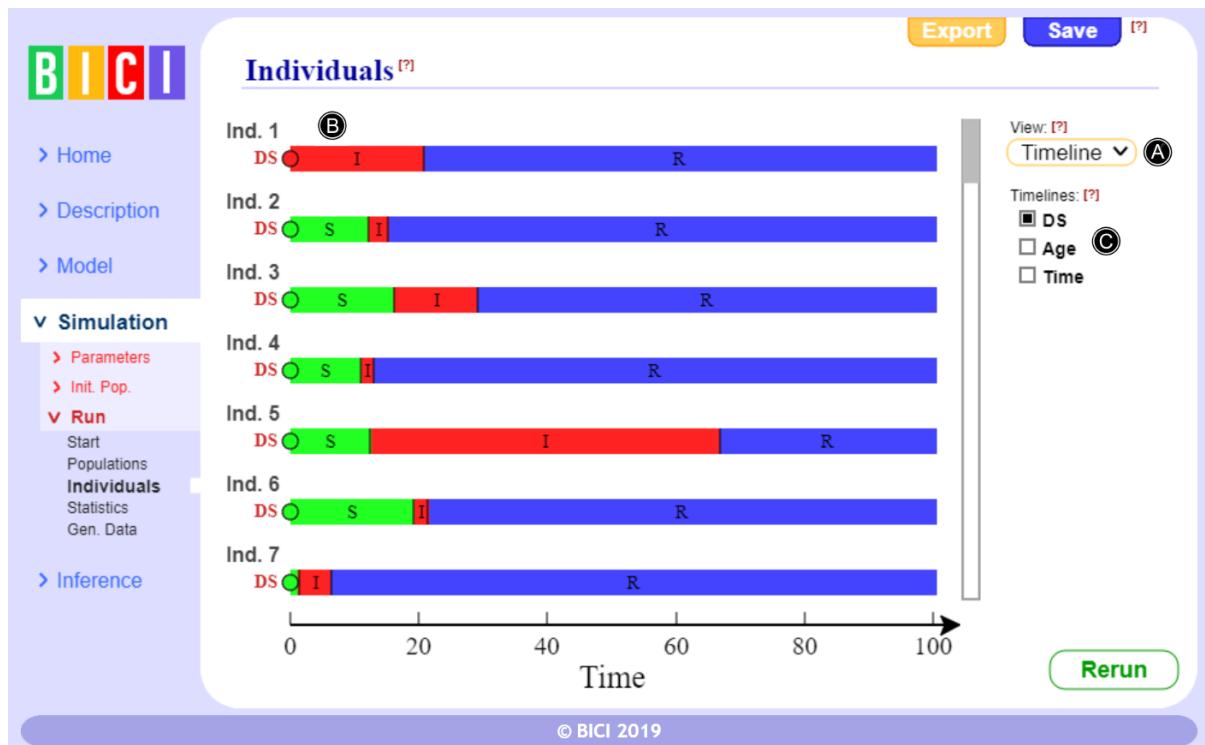


Figure 11 – Individuals: timeline view. Shows individual transitions for SIR model (EX 1). A: Select which view to use, B: Timelines for each of the individuals (different colours relate to different compartments in the classification), C: Select which classifications to display.

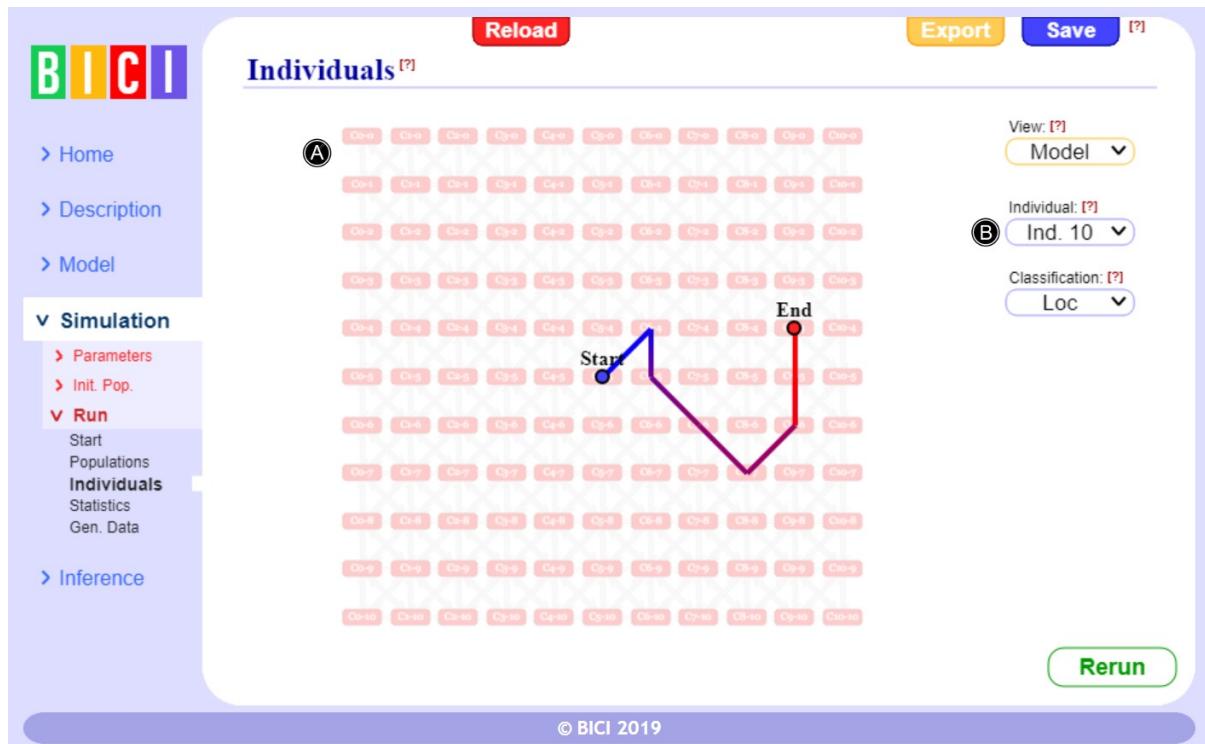


Figure 12 – Individuals: model view. A: Line showing the path of a selected individual on a spatial compartmental model (EX 30) with colour going from blue to red as a function of time, B: Select which individual to display for which classification.

3.3.2 Individuals

The ‘Simulation→Individuals’ page shows how individuals dynamically change state as a function of time. These variations can be viewed in two different ways (see option in Fig. 11A):

Timelines – Figure 11B shows an example taken from EX 1. Each individual in the population has a timeline showing when transitions in state occur. Note, the colours correspond to those defined in the compartmental model (green represents ‘S’, red represents ‘I’, and blue represents ‘R’). For models with multiple classifications, those actually displayed are selected using the checkboxes in Fig. 11C.

On the model – Example EX 30 models the geographical movement of individuals, where locations are discretised as an 11×11 grid of compartments. Figure 12A shows the movement of a particular individual (in this case ‘Ind. 10’ selected in Fig. 12B). The start and end points are indicated and the line follows the movement of the individual from blue to red as a function of time.

3.3.3 Statistics

The ‘Simulation→Statistics’ page summarises the parameter values used to run the simulations as well as other statistics.

The “Variable type” drop-down menu in the top right-hand corner allows the user to select different types of variable: First there is a list of all the classification names which contain parameters governing the transitions within these classifications, “Trans.” gives the number of transitions that occur, “Hyper.” gives information about hyper-parameters (if they are incorporated into the model) and, finally, “Misc.” gives the total number of individuals and transitions for the simulation as a whole.

When multiple simulations are carried out, statistics are given for both the mean and a range which encompasses 95% of the simulated values.

3.3.4 Generating data

On the ‘Simulation→Gen. Data’ page several options can be selected to generate hypothetical data based on simulated results (in fact these options were used to generate the datasets for the examples in §6). Hypothetical datasets can be used as inputs when performing inference. This not only provides a good test to show that inference is able to recover model parameters, but also is becomes an invaluable tool in estimating the power of a given data scenario.

The data consists of the infection (Dataset 1) and recovery (Dataset 2) times of individuals along with their initial infection status (Dataset 3). Simulated data were generated with 100 initially susceptible individuals and a single infected ($\beta=0.003$ and $\gamma=0.1$).

Name	Type	Class.	Time range	Data	Filter	Edit
(A) S → I	Transition	DS	0 — 100	Data	Filter	X
I → R	Transition	DS	0 — 100	Data	Filter	X
Init. DS	State	DS	0 — 0	Data	Obs. Mod.	X

(B) All individuals observed Unobserved individuals

(C) State Presence Transition Move
 Capture Capture ID Capture PD Population Derived

Next »

© BICI 2019

Figure 13 – Data sources page. A: List of all the sources of data currently loaded (taken from EX 1), B: Option if unobserved individuals exist in the system or not, C: Add a new data source.

Please edit entries if needed and press 'next' to continue.

ID	Time	Sex	Age	Ini
A02	43.2	M	10.3	S
A03	23.6	F	23.4	S
A04	10.3	M	5.6	S
A05	23.4	F	34.3	S
A06	15.3	F	22.3	S
A07	63.4	F	45.3	S
A08	20.4	M	30.4	S
A10	19.4	F	10.9	S
A11	49.3	F	18.4	S
A12	8.9	M	8.5	S
A13	22.4	M	23.4	S
A14	19.5	M	16.7	S
A15	34.5	F	24.6	S
A16	29.3	F	20.3	S
A17	34.2	F	18.4	S

Rows:16 # Cols:7

« Back Cancel Next »

© BICI 2019

Figure 14 – Loading data tables. A: Tables are loaded in '.txt' or '.csv' format and the relevant data is extracted using BICI's interface, B: Basic editing/manipulation of data can also be performed.

4 Inference

Bayesian inference is the process of combining data (which can take a variety of forms and may be inherently noisy) with previous knowledge regarding model parameters (the prior) to generate the best available estimates for model parameters and system dynamics (the posterior). The data, prior, and posterior are the subject of the next three subsections:

4.1 Data

Rather than loading all the data at once, the user loads different sources of data one at a time (in any order). As illustrated in Fig. 13, the ‘Inference→Data→Sources’ page shows the data sources currently loaded. This example is taken from EX 1 and shows three such sources (Fig. 13A): ‘S→I’ gives the timings of individual infections, ‘I→R’ gives the timings of individual recoveries and ‘Init. DS’ provides the initial disease status of individuals. The raw data can be viewed by clicking on the “Data” buttons (the “Filter” and “Obs. Model” options are discussed later). Data sources can be deleted by clicking on the corresponding red crosses on the right-hand side of this page.

The option in Fig. 13B selects whether collectively the data provides observations on all the individuals in the system or not. For example if there is only population level data (as in EX 19) or from capture campaigns (where not all individuals are necessarily caught, as in EX 21) the “unobserved individuals” option should be selected.

A new data source is added by selecting from the options at the bottom of the data sources page (Fig. 13C). Once clicked, the user will be prompted to either make use of a previously loaded data file or load a new one.

Data in BICI is loaded in either a tab delimited text (.txt) or comma-separated values (.csv) format, both of which can be exported from any spreadsheet software (Fig. 15 shows an example in Excel). The file must consist of a table (with headings) which contains columns related to the particular data source being added (as well as potentially other columns not used). Once loaded the user is requested to specify these relevant columns (Fig. 14A). BICI also provides some basic data editing capabilities (Fig. 14B) which allow searching, replacing, sorting, deleting (e.g. for removing missing data) and joining columns together.

The example in Fig. 14 shows data giving the infection times of individuals. This has been loaded by using the following procedure: 1) Click on “⊕ Transition” button (Fig. 13C), 2) load the ‘.csv’ file exported from the spreadsheet in Fig. 15, 3) select the column ‘ID’ giving individual identifiers, 4) select the column ‘It’ giving infection times, and 5) click on the ‘It’ heading and delete all blank entries (to remove individuals for which no infection is observed).

	A	B	C	D	E	F	G
1	ID	Sex	It	Rt	Weight	Age	Init. DS
2	A01	M		40.3	16.3	23.4	I
3	A02	M	43.2	54	15.4	10.3	S
4	A03	F	23.6	45.3	17.1	23.4	S
5	A04	M	10.3	53.2	17.3	5.6	S
6	A05	F	23.4	34.5	16.4	34.3	S
7	A06	F	15.3	29	20.1	22.3	S
8	A07	F	63.4		18.5	45.3	S
9	A08	M	20.4	54.3	19.4	30.4	S
10	A09	M			15.3	20.3	S
11	A10	F	19.4	54.3	12.4	10.9	S
	:				:		:

Figure 15 – Excel spreadsheet. Example epidemiological data with columns: ‘ID’ individual identifier, ‘Sex’ male or female, ‘It’ infection time, ‘Rt’ recovery time, ‘Weight’ weight in grams, ‘Age’ age in weeks and ‘Init. DS’ initial disease status (susceptible S or infected I). Blank entries correspond to infections and/or recoveries not taking place. This data needs to first be saved in ‘.txt’ or ‘.csv’ format to be loadable into BICI.

BICI

- > Home
- > Description
- > Model
- > Simulation
- < Inference**
 - < Data**
 - Sources
 - Individuals
 - > Prior
 - > Posterior

Observation Model

Here we define the probability of observing the classifier given a particular compartmental state.

Type: **Single** Multiple Diagnostic test General

Associates a given data value D with a given state.

Select compartment:

(A) D="I" [I] More...
(B) D="R" [R] More...
(C) D="S" [S] More...

Cancel **Done**

© BICI 2019

Figure 16 – Observation model for state data. A: This example is for the ‘single’ observation model (EX 4), B: Shows which data values D in the loaded table relate to which compartments in the model.

BICI

- > Home
- > Description
- > Model
- > Simulation
- < Inference**
 - < Data**
 - Sources
 - Individuals
 - > Prior
 - > Posterior

Observation Model

Here we define the probability of observing the classifier given a particular compartmental state.

Type: Single Multiple **Diagnostic test** General

Diagnostic test data assuming the test is sensitive to a defined set of compartments.
A sensitivity Se and specificity Sp account for inaccuracies in the test.

(A) Test sensitive to: **S X** **I ✓** **R X**

(B) Test name: Bl

(C) D="0" [+] +ve Test result -ve Test result No test
 $\Pr(D|S) = [Sp(Bl)]$ $\Pr(D|I) = 1-[Se(Bl)]$ $\Pr(D|R) = [Sp(Bl)]$

D="1" [+] +ve Test result -ve Test result No test
 $\Pr(D|S) = 1-[Sp(Bl)]$ $\Pr(D|I) = [Se(Bl)]$ $\Pr(D|R) = 1-[Sp(Bl)]$

Back **Cancel** **Done**

© BICI 2019

Figure 17 – Diagnostic test observation model. A: Define which compartmental states the test is sensitive to, B: Name of test, C: Shows how data values D in the loaded table correspond to either a positive (+ve), negative (-ve) or unknown test result. In this example D="0" and D="1" are interpreted as –ve and +ve results, respectively.

We now outline the data requirements and options for the various data source types (Fig. 13C):

4.1.1 State data

For a given classification, state data provides information (certain or uncertain) regarding the compartments individuals are in at specified points in time.

Three columns are required: Individual ID, time of observation and data value. Once a data table providing these three columns has been loaded, the user presses “Next”. The classification the data refers to is selected and this brings up the observation model (Fig. 16). Four different options are available depending on the nature of the data values:

Single – This associates a given data value with a given compartment. For example in Fig. 16B (taken from EX 4) the potential data values D=“I”, “R”, or “S” are interpreted in the observation model as individuals being in the ‘I’, ‘R’, or ‘S’ compartmental states, respectively. Note, however, that the values for the data and the names of the compartments don’t necessarily have to be the same. For example, the data may contain the values D=“Male” or “Female”, and these could be interpreted in the observation model as corresponding to the ‘M’ and ‘F’ compartments, respectively.

Multiple – This option is used when some data values are consistent with multiple compartmental states. For example, a data value D=“Infected” is consistent with both the ‘E’ and ‘I’ states in an SEIR compartmental model. The observation model allows the user to select (tick) the compartmental states consistent with a given observed data value.

Diagnostic test – This option provides (uncertain) information on whether individuals are in one set of states as opposed to another set of states. The most notable application of this is for binary disease diagnostic test results (see Fig. 17 taken from EX 5), which are routinely used to assess whether individuals are infected with disease or not. Such tests are typically imperfect, as characterised by a sensitivity Se (probability a truly infected individual tests positive) and specificity Sp (probability a truly uninfected individual tests negative). First the user specifies which compartment the test is sensitive to (in Fig. 17A this is simply the infected ‘I’ state, but for an SEIR model this may encompass both the ‘E’ and ‘I’ states. The name for the test is set (Fig. 17B), which is used as a subscript for the Se and Sp model parameters incorporated into the model. Finally, the data value is interpreted as representing either positive or negative test results or providing no information (Fig. 17C).

General – BICI provides the flexibility for the user to enter their own equations (see §2.2) to create the observation model that defines the probability of observing data values D given different underlying compartmental states (note, these expressions can contain parameters that themselves can be estimated during inference).

4.1.2 Presence data

This data type is simply used to inform BICI that certain individuals are present in the population at given time points (*e.g.* EX 21). Note, presence data is not required if state data is already available at those time points. Two columns are required: Individual ID and time of observation.

BICI

- > Home
- > Description
- > Model
- > Simulation

Inference

- Data**
- Sources
- Individuals
- > Prior
- > Posterior

Select filter

This data provides information about the transition: **S → I**

A Population: **Entire population** **Subpopulation**
The selected transition is observed for individuals in all compartments.

B Detection: **All transitions** **Some transitions**
All transitions of selected type are observed.

C Times: From **0** to **100**

Import [?]

Cancel **Done**

© BICI 2019

Figure 18 – Transition data filter. A: Define the population on which transitions are observed, B: Determines if all transitions of the specified type are observed or not, C: Time range over which specified transitions are observed.

BICI

- > Home
- > Description
- > Model
- > Simulation

Inference

- Data**
- Sources
- Individuals
- > Prior
- > Posterior

Detection probability

A Detection: **All individuals** **Individuals sampled**
Not all individuals are observed during a capture.

Please select whether the detection probability is the same or different across captures:

B **The same** **Different**
Individuals are captured with the following probability: **[p]** **(C)**

Import [?]

« Back **Done**

© BICI 2019

Figure 19 – Capture detection probability. A: Determines if all individual in a capture campaign are observed or not, B: Determines if the capture probability is the same across campaigns or not, C: Specifies capture probability.

4.1.3 Transition data

This type of data provides information about the timings of individual transitions. These are events in which either the compartment in one of the classifications changes (*e.g.* a susceptible individual becomes infected) or an individual enters (source) or leaves (sink) the system.

Two columns are required: Individual ID and time of transition. Once the data is loaded, the user specifies which transition these observations relate to and the observation “filter” used. This filter is specified in three ways (see Fig. 18): Firstly, Fig. 18A determines if the entire population is observed or only a subpopulation, *e.g.* at a particular location. Secondly, Fig. 18B selects whether all the specified transitions on the specified population are observed or only a fraction (this fraction can be incorporated as an estimable parameter in the model, *e.g.* EX 23). Finally, Fig. 18C specifies the time range over which observations are made. All observed transition times in the data must lie within this range.

4.1.4 Move data

This is like transition data, except that here changes in state are not as a result of processes within the model itself but externally imposed (consequently move data has no likelihood associated with it). For example, it could be used to incorporate the physical movement of animals from one location to another by lorry or the effect of culling individuals in an infected wildlife population.

Two columns are required: Individual ID and time of move. Unlike transition data, no filter is needed.

4.1.5 Capture data

A “capture campaign” refers to stochastically sampling (*e.g.* observing individuals with a capture probability p) a population (or subpopulation) at a given point in time. In the ecological setting this corresponds to setting traps to catch a proportion of the overall population of animals which are then marked and released back into the wild. Capture-mark-recapture studies make use of successive capture campaigns to study wildlife groups, *e.g.* see EX 36.

It is important to note that data from individuals actually captured during capture campaigns is incorporated either as 'state' data or simply 'presence' data. The 'capture data', referred to here, provides information on where and when capture campaigns themselves are carried out, and the probability of capturing individuals during these campaigns.

Three columns are required: Capture campaign name (which must be unique), the population being sampled from, and the time of the campaign. If the entire population is being sampled from then ‘All’ is used in the middle column, else the specific compartment is stated *e.g.* location (also compartments from multiple classifications can be specified separated by commas *e.g.* ‘Loc1,M’ means sampling of the male population at location ‘Loc1’).

Figure 19 specifies the capture probability for capture campaigns in EX 23. Firstly, Fig. 19A determines if all individuals are observed or not. Secondly, Fig. 19B specifies if all capture campaigns have the same detection probability or not (if not, ‘capture PD’ data is required, see below). Finally, Fig. 19C gives an editable expression for the capture probability, which in this case is determined by model parameter p that is estimated during inference.

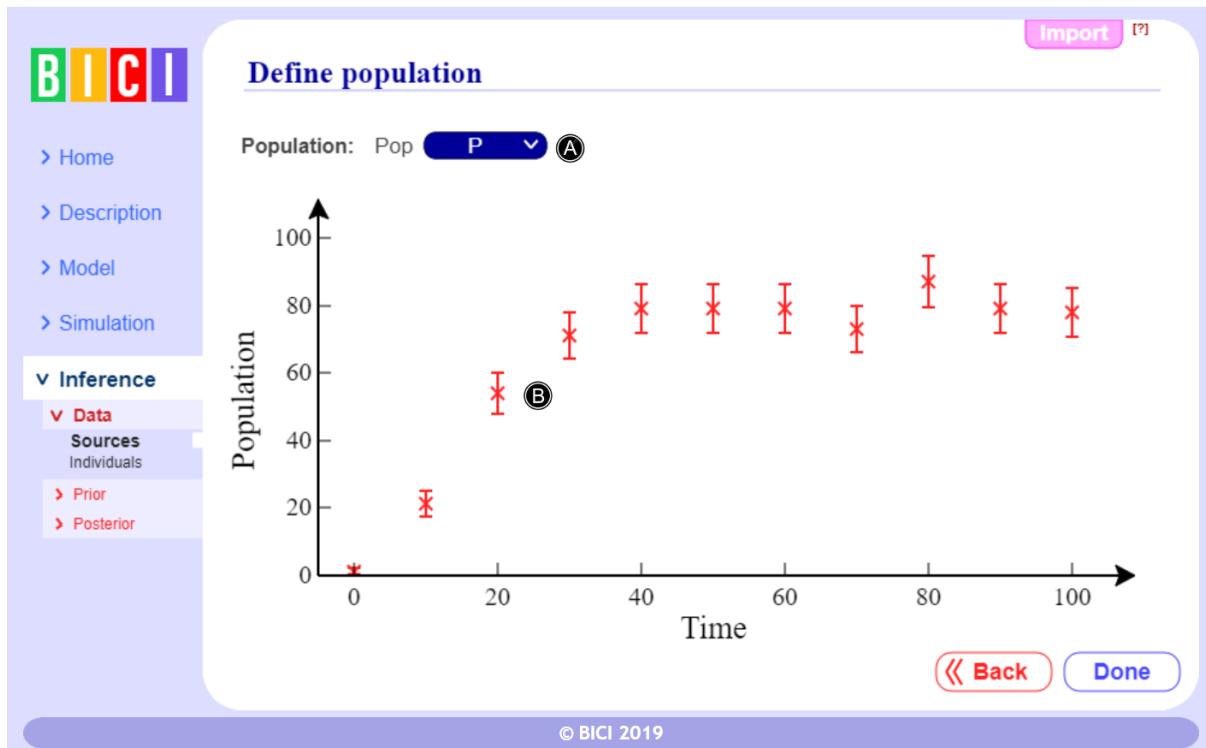


Figure 20 – Population data. A: Determines population/subpopulation being measured, B: Graphically shows the population estimates along with 95% confidence intervals reflecting uncertainty in these estimates.

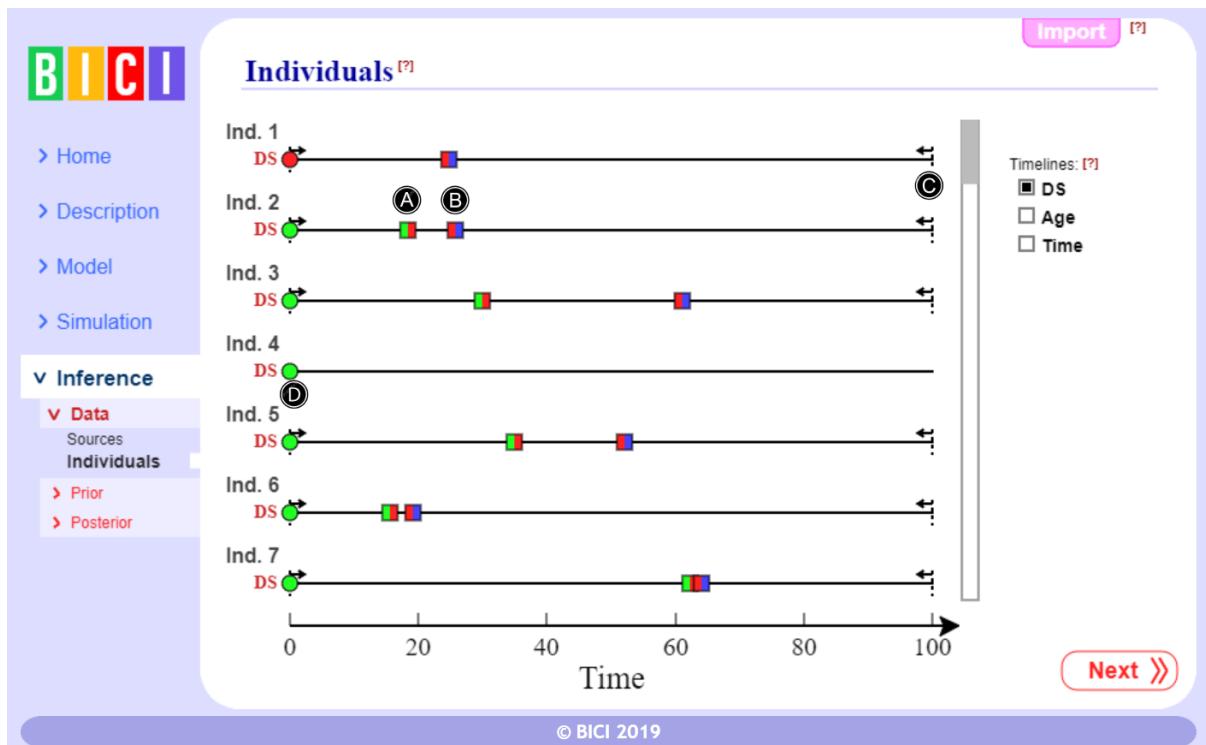


Figure 21 – Individual data. Summarises all individual-based data as symbols on individual timelines (EX 1). A: Infection times, B: Recovery times, C: Time range of transition observations, D: Initial disease status.

In reality a capture campaign may take some period of time to carry out (*e.g.* animal traps may be laid over several consecutive days). For analysis in BICI, however, it is necessary to specify a single time point (*i.e.* at the middle of the capture campaign), which is also used in any state data obtained from individuals that were caught.

4.1.6 Capture ID data

When capture campaigns are carried out it is assumed that if an individual is observed at the same time as the campaign (in state or presence data), it has been caught in that campaign. However in some circumstances capture campaigns are performed concurrently (*e.g.* at different locations at the same time). To clarify this 'capture ID' data is used to specify which individuals are observed in which capture campaigns.

Two columns are required: Individual ID and name of capture campaign.

4.1.7 Capture PD data

Usually capture campaigns have associated with them a universal capture probability (given by an estimable parameter, as in Fig. 19C). If, however, this is not the case, the option in Fig. 19B can be set to 'Different' and 'Capture PD' data is added to allow the capture probability to be specified separately for each capture campaign.

Two columns are required: Capture campaign name and equation (see §2.2) giving the probability of detection (which may include estimable parameters).

4.1.8 Population data

This type of data estimates population or subpopulation sizes at different points in time. Three columns are required: The time, the estimated size of the population and a standard deviation in that estimate to represent its uncertainty.

Figure 20 shows an example taken from EX 19. Figure 20A specifies the population or subpopulation being estimated (in this case the size of the entire wildlife reservoir P) and Fig. 20B visually shows the population estimates along with the uncertainties associated with them (the observation model assumes a gamma distributed error with mean given by the population estimate and specified standard deviation).

4.1.9 Derived data

This is analogous to population data except that rather than estimating population sizes it estimates the size of a specified derived quantity in the model (see §2.6). Three columns are required: The time, the estimated size of the derived quantity and a standard deviation in that estimate (to represent uncertainty).

Example EX 10 provides an example of this type of data in which estimates of environmental pathogen level provide indirect information regarding the number of infected individuals in the populations (through an estimated proportionality constant).

Because derived quantities can be negative as well as positive, a normally distributed (instead of gamma distributed) observation model is used with mean given by the estimate and specified standard deviation.

4.1.10 Individuals

The ‘Inference→Data→Individuals’ page graphically summarises all individual-based data. Figure 21 shows an example taken from EX 1. The susceptible, infected and recovered compartments are colour-coded green, red, and blue, respectively. Consequently, infection transitions are represented by green/red squares (Fig. 21A) and recovery transitions by red/blue squares (Fig. 21B). The time range over which transitions are observed is denoted by two vertical dashed lines with inward pointing arrows (Fig. 21C). Finally, the initial disease status of individuals is given by the green or red circles at time $t=0$ (Fig. 21D).

Figure 22 provides a key for all the different types of individual-based data that can be loaded into BICI.

- State data (single): in red compartment
- State data (multiple): in orange/red/blue compartment
- State data (diagnostic test): -ve result
- State data (diagnostic test): +ve result
- State data (diagnostic test): two test results
- Transition data: from green to red compartment
- Transition data: observed time range

Figure 22 – Key showing symbol types. These data types are described in sections 4.1.1-3.

Prior

A Defined [?] α_{Group} Normal Mean: 0 SD: $[\sigma]$

B Compartmental model [?] β Flat Min.: 0 Max.: 1 σ Flat Min.: 0 Max.: 1

C Observation model [?] Se(Blood) Fix Value: 0.5 Sp(Blood) Fix Value: 0.95

D Initial state [?] $\xi_{\text{Group,DS}}$ Dirichlet $\alpha: 1$

Import [?]

Next >>

Figure 23 – The prior. This page specifies the prior for different aspects of the model (EX 16). A: Shows prior distributions placed on model parameters, B: Priors on parameters in the compartmental model as well as hyperparameters, C: Priors on parameters in the observation model, D: The Dirichlet prior placed on individual initial states.

4.2 Prior

The prior is used to capture knowledge regarding parameter values before the data itself is considered. For example, often on physical grounds it is possible to isolate a range of plausible values for a particular parameter onto which a flat prior distribution can be placed. BICI requires that all model parameters have proper priors (*i.e.* priors that are bounded such that they integrate over the entire parameter range to one).

Figure 23 shows the ‘Inference→Prior’ page used to set priors. BICI supports the following prior specifications: flat, which relates to a uniform probability distribution across a range, and gamma, normal, log-normal, exponential, beta, and Weibull distributions, as well as the possibility to fix parameters to specific known values.

For convenience parameters are grouped by type:

Defined – This shows model parameters assumed to be drawn from distributions defined within the model itself (Fig. 23A shows a case from EX 16, in which the parameters α_{Group} are assumed to be random effects). These distributions can be edited by going to the ‘Model→Distributions’ page (see §2.6 for details) .

Compartmental model - These are priors on parameters used in the compartmental model along with any hyperparameters. In Fig. 23B this includes not only the transmission rate β , but also σ which defines the standard deviation of the normal distribution from which α_{Group} is drawn.

Observation model - These are priors on parameters used in the observation model (which relates data measurements to underlying system dynamics). In EX 16 the sensitivity Se and specificity Sp are fixed to known values (Fig. 23C).

Initial state - The compartmental state for those individuals present at the start time is assumed to take a Dirichlet distribution (Fig. 23D), where the α values set the relative probability of being in a given state. By default, all α values are set to one representing a flat prior (as shown in Fig. 23D), but an SIR model with $\alpha_S=2$, $\alpha_I=1$, $\alpha_R=1$ would correspond to a prior belief that individuals are twice as likely to initially be in the susceptible ‘S’ state than in the other two states.

Additionally, smoothing priors can be placed on parameters that are age or time dependent (through the ‘Age’ and ‘Time’ model classifications in §2.4 and §2.5). An example of this is given in EX 22, where the mortality rate μ_{Age} of individuals is estimated as a function of 6 age classifications. The basic idea is that often we expect the underlying variation with age (or time) to be smooth (*e.g.* juveniles and older individuals will tend to have a higher mortality rate and middle aged individuals will tend to have a lower mortality rate leading to an overall smooth U-shaped profile). This prior expectation is incorporated into the prior by introducing a penalty for large variation in μ_{Age} between consecutive age compartments.

Three options are available for age (and also time) smoothing: ‘None’ introduces no smoothing, ‘Smooth’ adds a contribution $-\sum_{a=1}^{A-1} \nu(p_{a+1} - p_a)^2$ to the log of the prior probability, where A is the number of age compartments, p_a is a model parameter that depends on age a , and ν is a smoothing parameter specified in the prior, and ‘Log Smooth’ which adds a similar contribution for the log of the parameter values $-\sum_{a=1}^{A-1} \nu(\log(p_{a+1}) - \log(p_a))^2$ (note, this last option is

appropriate only when p_a is strictly positive for all a). The advantage of the ‘Log Smooth’ option is that v is dimensionless quantity, and so a value around 10 will work well under most circumstances (in effect this option smooths the *percentage* change in p_a between successive age compartments). In contrast for the ‘Smooth’ option v is not dimensionless, hence its size must be carefully chosen depending on the expected magnitude of p_a .

4.3 Posterior

Based on the data entered in §4.1 and prior knowledge from §4.2 it is generally not possible to identify model parameters with perfect precision (or transition events for that matter, unless they are specified by the data). Rather, there exists a distribution in these quantities known as the “posterior”, which expresses both a best guess for parameters (*i.e.* posterior mean) and a range in values consistent with the data (*i.e.* credible interval). BICI achieves Bayesian inference by means of drawing samples from the posterior distribution using a widely applied technique known as “Markov chain Monte Carlo” (MCMC). Unlike many other statistical techniques (such as maximum likelihood), MCMC does not simply follow a predetermined algorithm and produce a final end result. Instead, it successively generates samples that progressively improve the accuracy with which the posterior is approximated. MCMC is continuously run until sufficient precision is achieved (determining how long this process should actually take is discussed in §4.3.8).

4.3.1 Starting inference

The page ‘Inference→Posterior→Start’ provides several options that must be selected before inference can start:

- **MCMC runs** – An MCMC run consists of a random initialisation step (which sets the initial model parameters and events⁴) followed by the MCMC iteration procedure outlined above. The number of parallel MCMC runs to be executed is selected using a drop-down menu. Each run exists on the computer as a separate process, and since most modern computers contain multiple CPU cores, computationally efficiency can be substantially improved by selecting more than one run⁵. Additionally, executing multiple runs allows for MCMC diagnostics to verify convergence (see §4.3.8). A suitable choice is 3 MCMC runs, which is used as the default setting.
- **Time range** – The time range over which inference is performed is set. This may just include the period over which data is collected, but can be extended forwards and backwards in time to predict future and past dynamic behaviour, *e.g.* see EX 6.

⁴ Note, in BICI the initial state is not a truly random sample from the model, because it must be consistent with the prior and data (*i.e.* have a non-zero prior and observation model probability).

⁵ Executing more runs than the number of CPU cores can lead to a substantial slowing down of the BICI interface. This is not expected to provide any further improvement in computational efficiency, and so is not recommended.

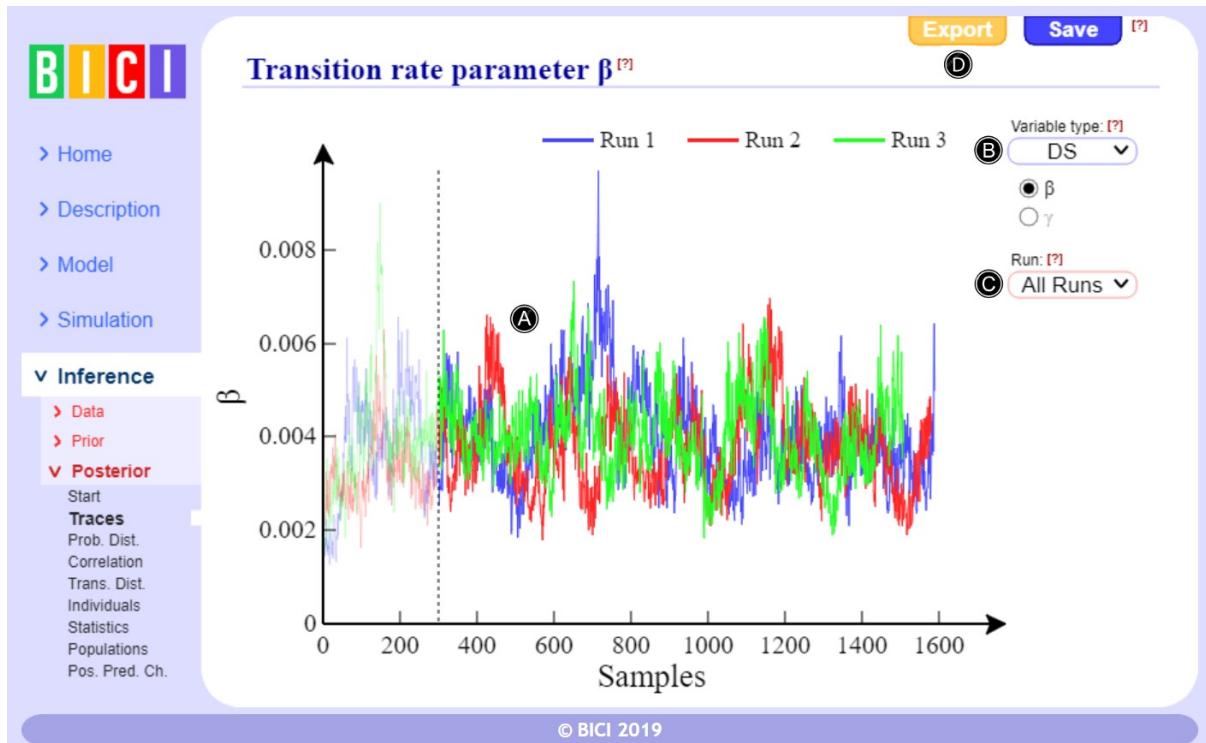


Figure 24 – Trace plot. A: Shows posterior samples for the transmission rate parameter β as MCMC is iterated (taken from EX 5) for 3 independent MCMC runs (colour coded), B: Select variable, C: Select which runs are displayed, D: Export output (see §5.2).

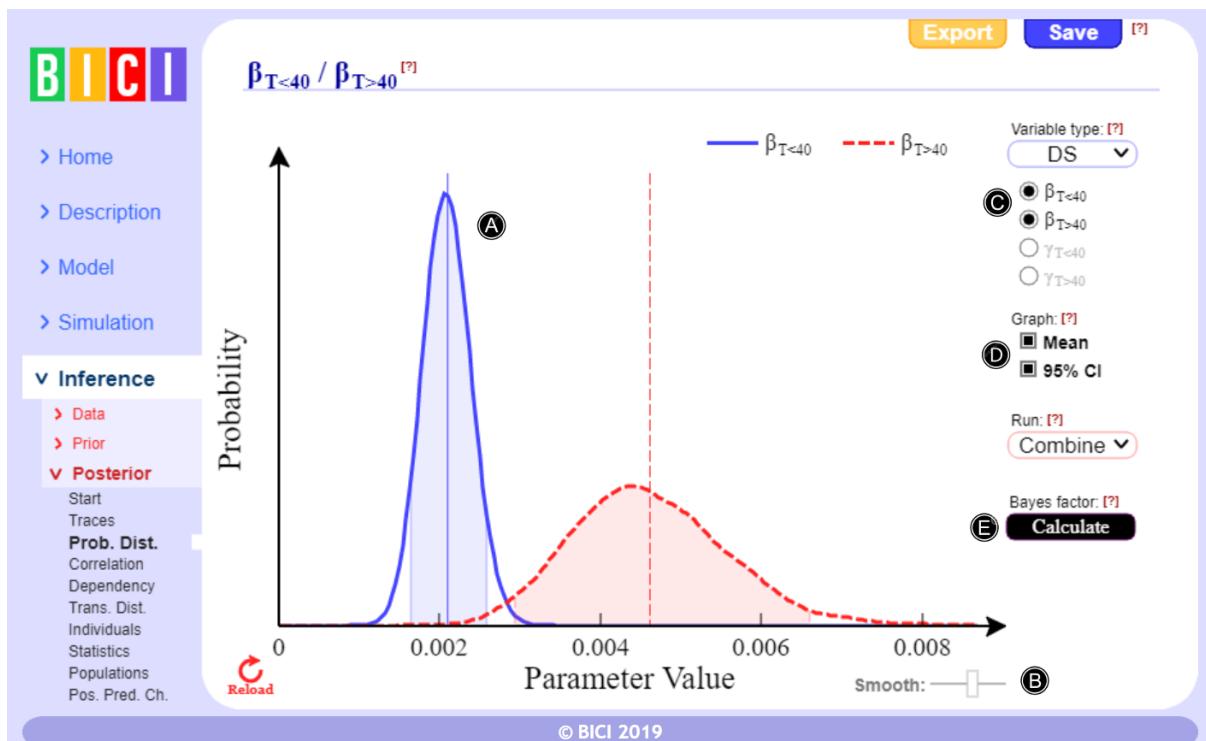


Figure 25 – Distributions. A: Probability distributions (use ctrl key to select multiple parameters) for two model parameters (taken from EX 7), B: KDE smoothing parameter, C: Parameter selection, D: Mean and credible interval options, E: Calculate the Bayes factor.

On the advanced options page further specifications can be made:

- **Sample limits** – The maximum number of stored parameter and event sequence samples is defined (when exceeded, BICI thins the existing samples by a factor of two and subsequently stores them at half the rate). Increasing these limits can lead to smoother, more accurate, posterior plots, but also large memory consumption (depending on the size and complexity of the model).
- **Individual limit** – The maximum number of individuals is set (by default this is 10^5). Such a limit is placed to avoid BICI crashing due to excessive memory usage.
- **Termination** – Three option are available regarding inference termination: either it continues indefinitely until manually stopped, a fixed number of MCMC updates are performed, or termination happens when convergence is achieved (this occurs when every model parameter has an effective sample size that exceeds a specified threshold and a Gelman-Rubin statistic that is below a specified threshold, see §4.3.8 for details).

Bayesian inference can now begin by clicking on “Start” button (Fig. 9E). The following sections describe different ways in which the posterior can be analysed/visualised by selecting different tabs on the ‘Inference→Posterior’ submenu.

4.3.2 Trace plots

The first page that appears after inference is started shows trace plots for model parameters (Fig. 24). As mentioned previously, MCMC works by successively drawing parameter samples (represented by the x -axis) from the posterior. Ideally these samples should be randomly distributed across the posterior, but in reality they are correlated (which manifests itself as structure within these plots).

So-called “mixing” describes the degree to which the samples generated so far are representative of the posterior as a whole. The example in Fig. 24 is one in which mixing is relatively good, because the curves exhibit substantial variation up and down about the posterior mean. Under different circumstances, however, MCMC runs can exhibit poor mixing, resulting in BICI taking much longer to provide reliable results. The examples in §6 take from a few seconds to a few minutes to adequately mix. Measures for assessing how long inference should be performed before MCMC is stopped are discussed in §4.3.8.

A drop-down menu (Fig. 24B) allows the user to select different types of variable. First there is a list of all the classification names which contain parameters governing the transitions within these classifications, “Init. Prob.” gives parameters associated with the probability of an individual's initial state, “Trans.” gives the number of transitions that occur, “Hyper.” gives information about hyper-parameters, and “Misc.” gives other quantities, such as the overall posterior probability, the observation model, the latent process likelihoods and the prior as well as the total number of individuals and events. Below the drop-down menu (Fig. 24B) different parameters can be selected.

The vertical dotted red line in Fig. 24A represents the so-called “burn-in” period. Samples before this point are judged to be unduly influenced by the initial random starting configuration, and so they are discarded from subsequent analysis. To maximise efficiency this burn-in period is dynamically shifted as more and more posterior samples are generated (burn-in period is set to around 20% of the overall sample number).

4.3.3 Probability distributions

The raw posterior samples from the previous section can be converted into posterior probability distributions, as shown in Fig. 25. These are generated using a technique known as kernel density estimation (KDE) [2]. KDE makes use of a smoothing parameter which can be adjusted by means of a slider (Fig. 25B). This particular example shows distributions for two model parameters, which can be achieved by holding down the control key and sequentially selecting the relevant parameters, one at a time (Fig. 25C). The options in Fig. 25D select whether the means (vertical lines) or 95% credible intervals (shaded regions under the curves with unshaded tails) are displayed.

A Bayes factor (BF) is the ratio of the likelihood for one particular hypothesis to the likelihood for another [3]. The BF comparing the full model to one in which a particular parameter is fixed (usually to zero, such that the parameter is effectively removed from the model) can be calculated using the button in Fig. 25E. This is one way to determine which parameters in the model are redundant, so allowing the model to be simplified in a step-wise fashion. A BF between 3 and 10 represent moderate evidence for one model over another and exceeding 10 is considered strong evidence [4].

4.3.4 Correlation

Figure 26A shows posterior correlations between different model parameters. The parameters of interest are selected by clicking on the checkboxes on the right hand menu (Fig. 26B). The Pearson correlation matrix is then automatically calculated. Values near to one indicate a high degree of correlation (*i.e.* when one parameter is high in a given a posterior sample then most likely the other will also high in that same sample⁶). Values near to minus one indicate anti-correlation (*i.e.* when one parameter is high then most likely the other is low). Values near zero represent little correlation.

Clicking on one of the elements in the matrix displays a 2d posterior distribution for the corresponding variables, *e.g.* Fig. 27A shows the a KDE plot for the transmission rate β and incubation duration m parameters taken from the SEIR model in EX 9. Alternatively a scatter plot can be displayed by selecting from the options in Fig. 27B.

Visualising parameter correlations is useful for investigating and understanding confounding between different model parameters.

4.3.5 Dependency

Compartmental models often contain parameters that depend on other classifications within the model, *e.g.* in EX 15 the disease transmission rate β_{vac} depends on the vaccination status classification ‘Vac’, and in EX 22 the mortality rate μ_{Age} depends on the ‘Age’ classification.

If such parameters exist within the model, they can be visualised using the dependency page (see Fig. 28). For cases in which the dependent classification is either ‘Age’ or ‘Time’ the resultant plot has a continuous x -axis with crosses and error bars representing posterior means and 95% credible intervals. A continuous line joining the crosses is shown to approximate the true continuous underlying variation in the quantity. For other classifications the x -axis is divided equally into the various compartmental categories, with results shown separately for each.

⁶ Note, high and low are measured relative to a parameter’s own mean,

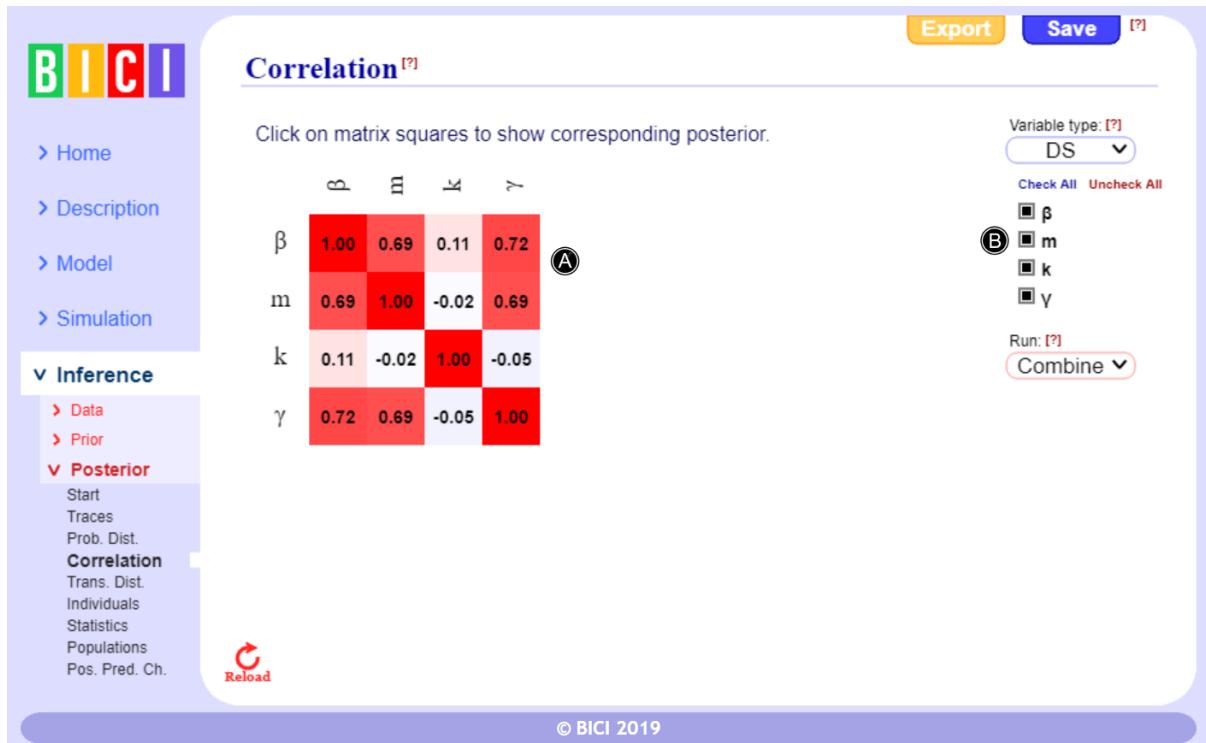


Figure 26 – Correlation matrix. A: Shows a matrix of Pearson correlation coefficients for different combinations of model variable (this is taken from example EX 9). Clicking on a matrix element leads to a corresponding 2d posterior plot for the two parameters, as shown in Fig. 27). B: Select parameters using the checkboxes.

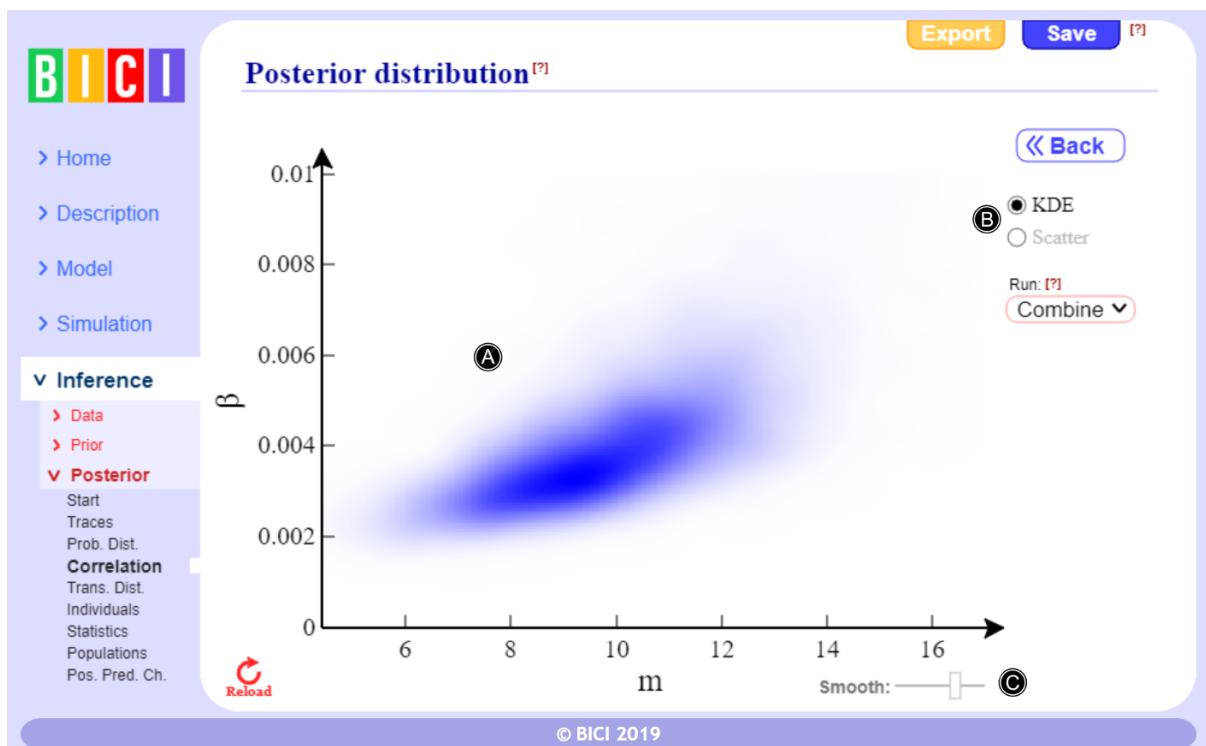


Figure 27 – 2d posterior plot. A: Shows marginalised posterior probability for two model parameters (the transmission rate β and the incubation duration m in EX 9), B: Select to display kernel density estimate (KDE) or scatter plot, C: Smoothing parameter in the case of KDE.

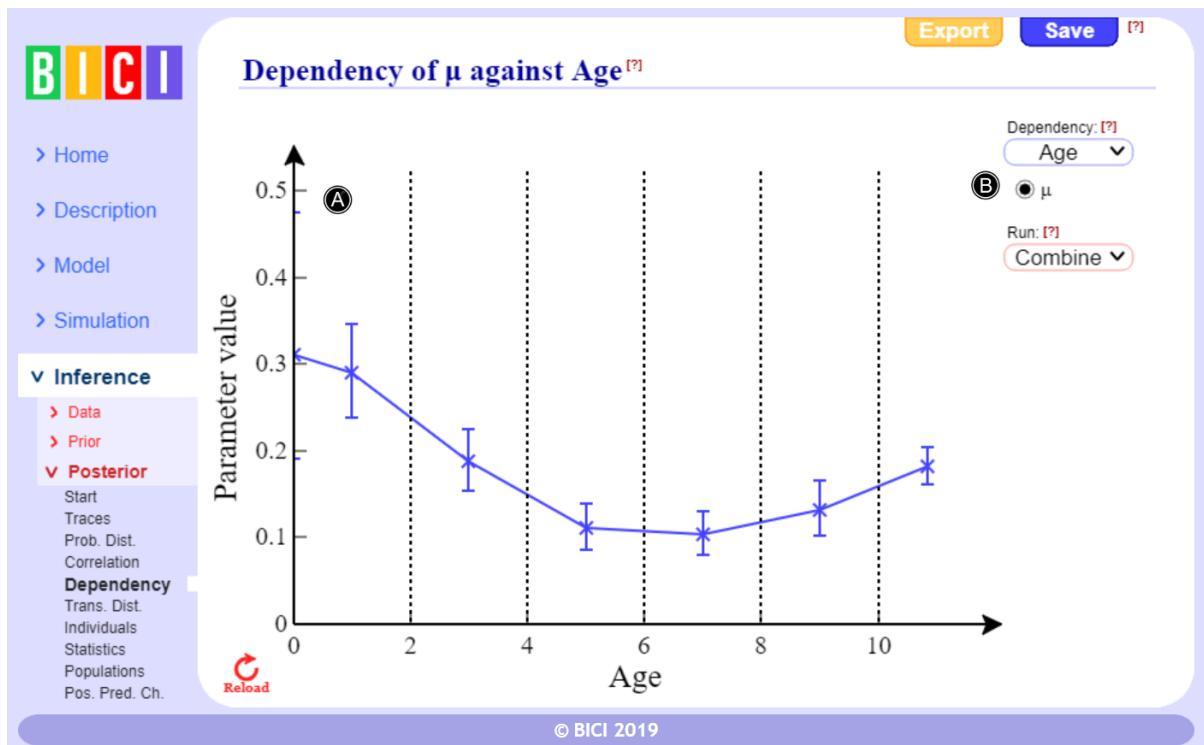


Figure 28 – Dependence plot. Displays model parameters with a dependency on a model classification. A: This example show how mortality depends in the ‘Age’ classification from EX 22, B: Parameter selection.

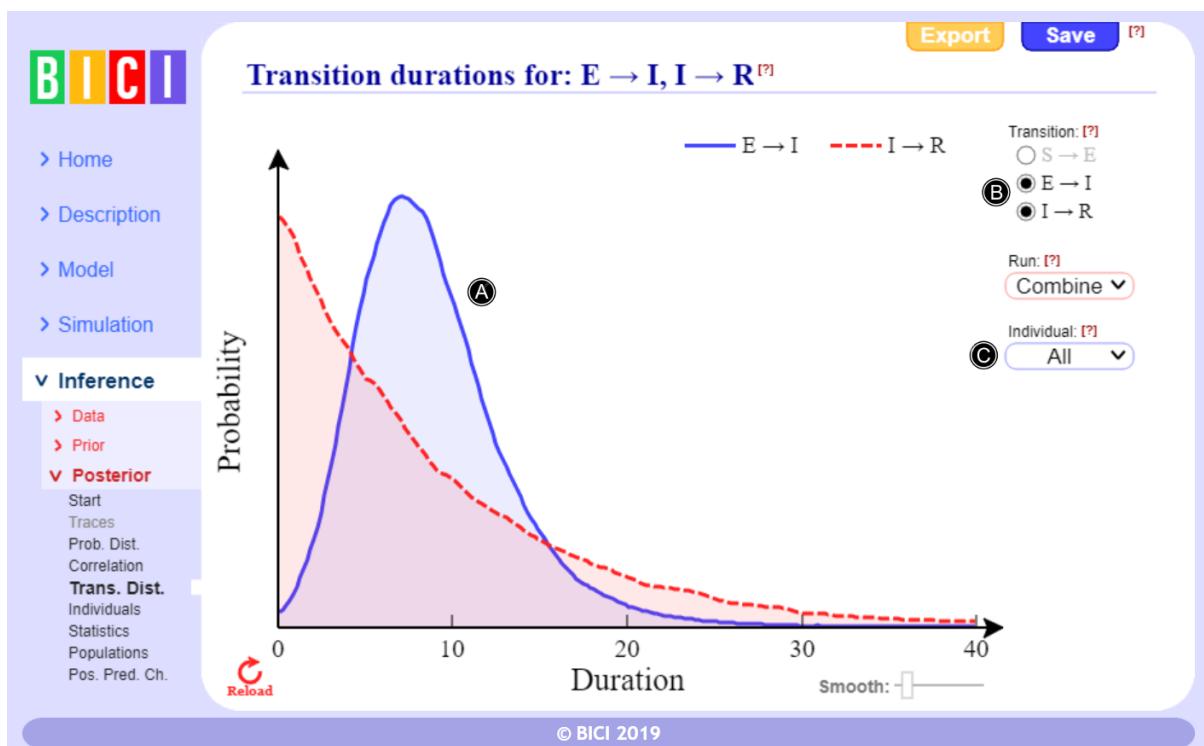


Figure 29 – Transition duration distribution. A: Shows posterior distribution(s) for transition durations. This example shows the incubation (blue) and recovery (red) times from EX 8. B: Transition selection (use ctrl key to select multiple transitions), C: Select specific individual.

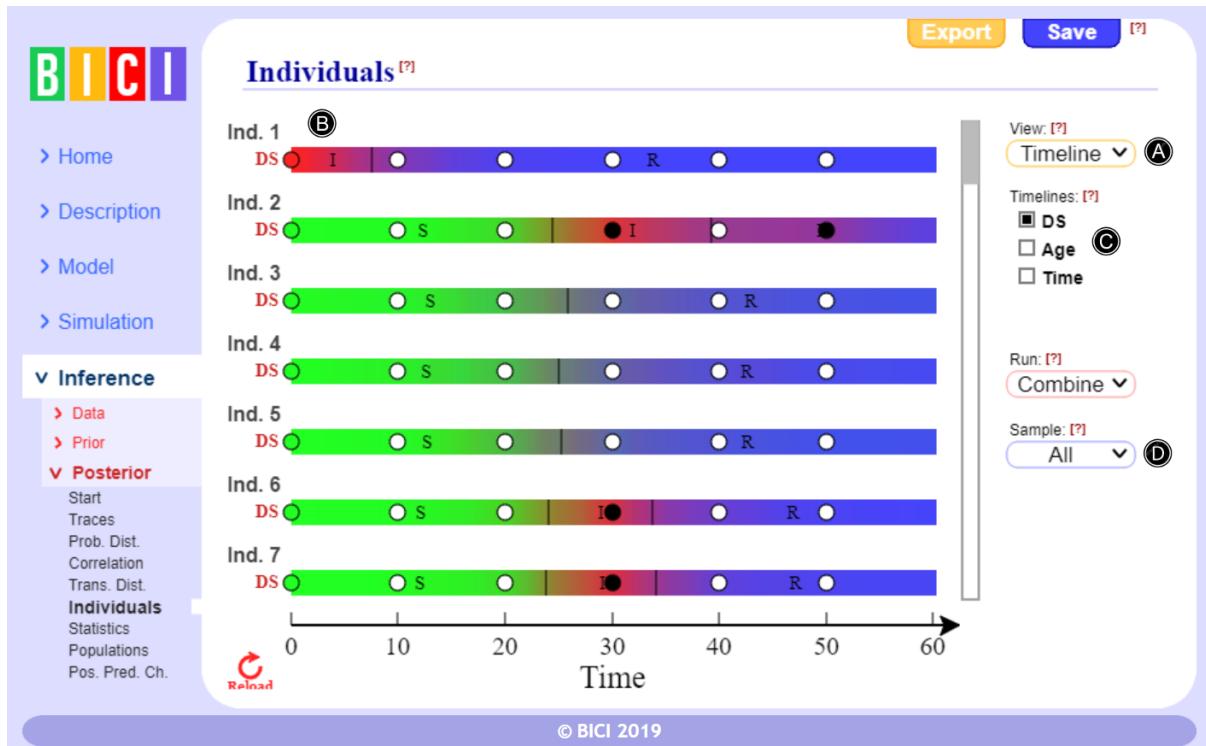


Figure 30 – Individuals: timeline view. A: Select which view to use, B: Timelines for each individual, where symbols relate to the data and gradations in colour relate to the probability of being in different compartments (from example EX 5), C: Select which classifications to display, D: Select specific posterior sample.

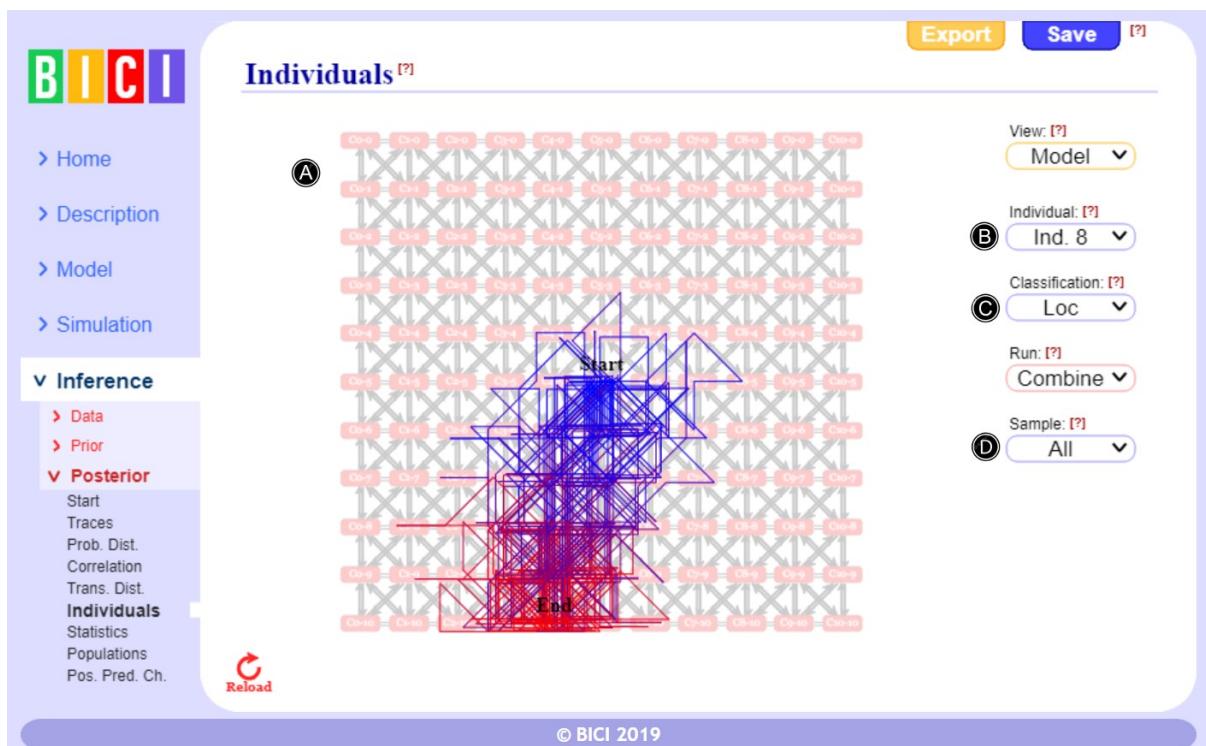


Figure 31 – Individuals: model view. A: Shows the posterior paths of a selected individual on the spatial compartmental model in EX 30 (lines go from blue at the start time to red at the end time), B: Select which individual to display, C: Select classification, D: Select specific posterior sample.

4.3.6 Transition distributions

As illustrated in Fig. 29A, this page shows estimated probability distributions for the duration of selected transitions (Fig. 29B), *i.e.* the distributions in the time an individual spends in the initial compartment before it moves to the final compartment. These distribution are generated from posterior samples using kernel density estimation with smoothing parameter adjusted by means of the slider in the bottom right hand corner. The graph can be generated for the population as a whole, or for just a specified individual (Fig. 29C).

Visualising posterior transition distributions provides a useful diagnostic tool to determine whether they conform to the distributions defined in the model itself. For example, a Markovian transition is expected to generate an exponentially distributed duration profile. If, however, the inferred duration profile (*i.e.* taking into account the data) is peaked, this suggests the model should be modified to incorporate a gamma distributed transition instead of Markovian.

4.3.7 Individual timelines

These show how the posterior probability for an individual's state dynamically changes as a function of time. These variations can be viewed in two different ways (see dropdown menu in Fig. 30A):

Timelines – Here each individual in the population has one or more timelines (Fig. 30B). The symbols (see Fig. 22 for reference) represent the data, which in this case consists of the initial disease status and subsequent periodic disease diagnostic test results (black/white circles denoting positive/negative results). The gradations in colour represent variation in posterior probability of being in the different compartmental states, *e.g.* pure green represents a high posterior probability the individual is in the susceptible 'S' state, but a colour between green and red represents posterior uncertainty between these two states. The annotations 'S', 'I' and 'R' (with corresponding black vertical dividing lines) in Fig. 30B denote time periods for which the named compartment has the highest posterior probability. Displayed classifications are selected using the checkboxes in Fig. 30C.

On the model – Example EX 30 models the geographical movement of individuals, where locations are discretised as an 11×11 grid of compartments. Figure 31A shows posterior samples for the movement of a particular individual (in this case 'Ind. 10' selected in Fig. 31B) on this landscape. The approximate start and end points are indicated and the lines following the movement of the individual change from blue to red as a function of time.

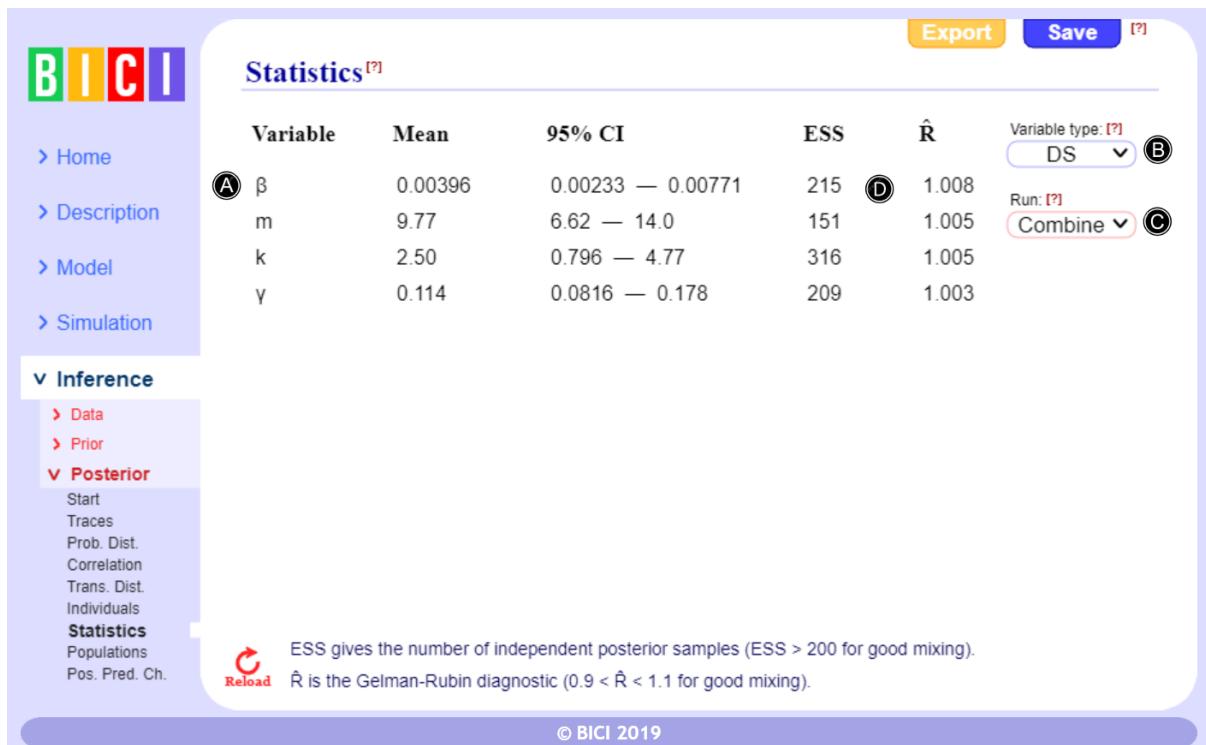


Figure 32 –Statistics. A: Summary of posterior parameter estimates (taken from EX 8), B: Variable type selections, C: MCMC run selection, D: MCMC diagnostics.

4.3.8 Statistics

As shown in Fig. 32A for the SEIR model used in EX 8, BICI summarises the posterior mean and 95% credible intervals for all model parameters. Various variable types can be selected from the drop-down menu in Fig. 32B, including those from the compartmental model, the observation model, transitions and miscellaneous quantities such as the posterior probability, latent process likelihood and prior. Estimates can be made combining the samples from all MCMC runs together (using the “Combine” option in Fig. 32C) or for just a specific run.

Two measures are used to test for MCMC convergence (Fig. 32D):

- **Effective sample size (ESS)** – This takes into account the fact that successive MCMC samples are correlated, and works out how the total sample number is reduced to give an estimate of the effective number of independent draws from the posterior. An ESS exceeding 200 is considered indicative of good mixing. Looking at the example in Fig. 32D we find that because the ESS for the incubation duration parameter m is only 151, this indicates that BICI needs to be run for longer to ensure reliable results. Note, the ESS is not guaranteed to monotonically increase (in fact if it is less than 100 it often fluctuates wildly). As its value increases, however, these fluctuations should dampen as a result of convergence towards the true posterior distribution.

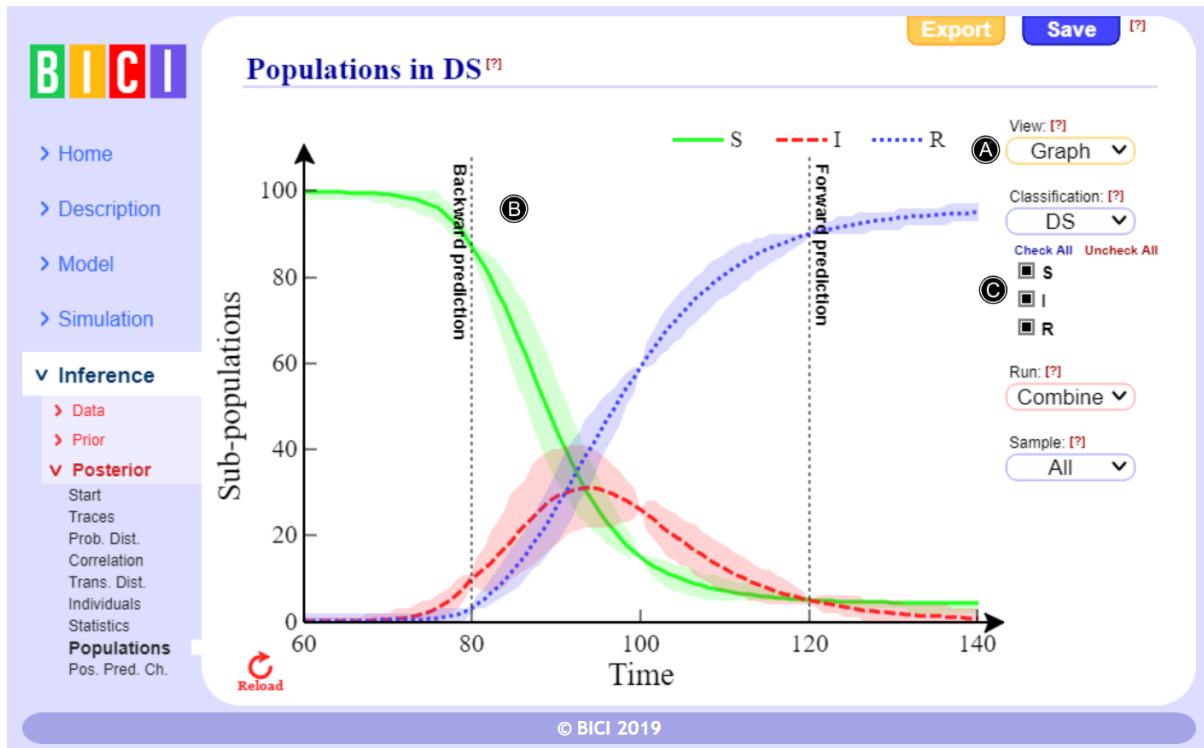


Figure 33 – Population: graph view. A: Select the type of view, B: Graph showing posterior population sizes as a function of time with shading representing 95% credible intervals (taken from EX 7), C: Choose which populations to show.

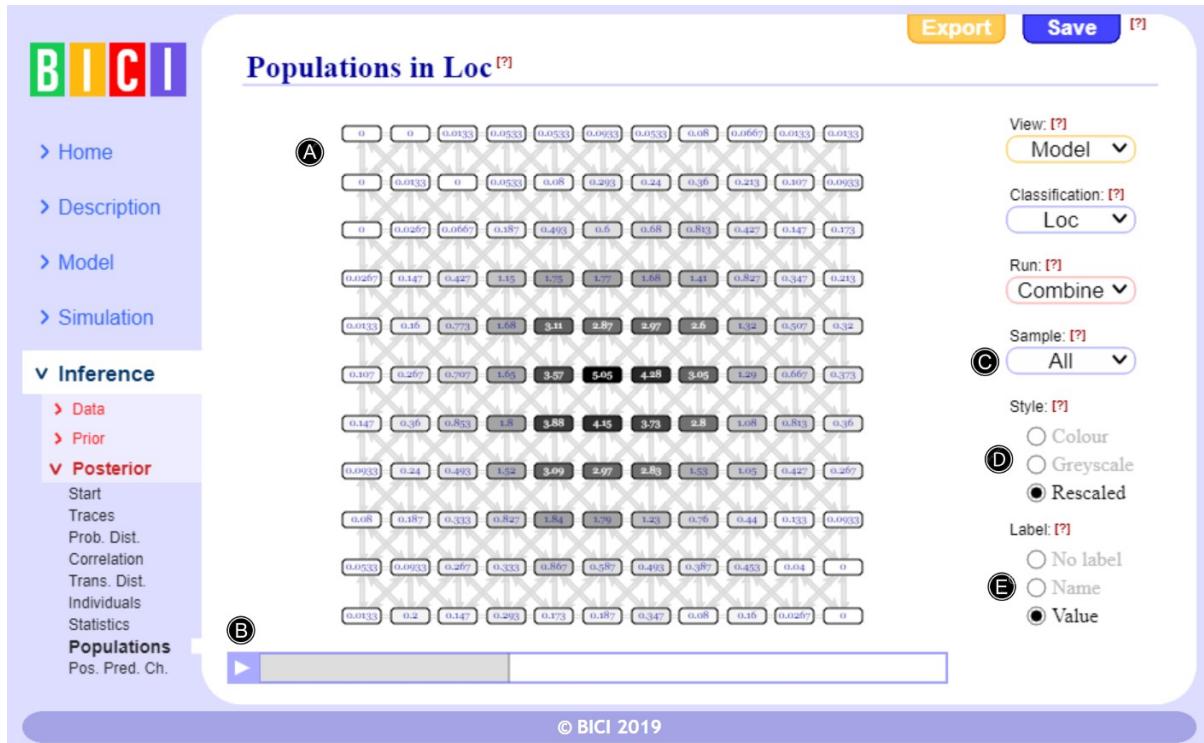


Figure 34 – Population: model view. A: Shows populations in compartments (for spatial model EX 30) with darker shading representing larger population size, B: Play button and time bar, C: Select a posterior sample or use all, D: Options for style, E: Options for label.

- **The Gelman-Rubin statistic \hat{R}** – On top of poor mixing, another difficulty faced by MCMC is multimodality. This refers to the situation in which one MCMC run ends up in one posterior mode (and perhaps mixes well) but another goes to a completely different mode⁷. One way to check for this problem is to use the Gelman–Rubin convergence diagnostic [5], which verifies whether multiple runs converge on the same mode or not. It involves calculating the so-called “potential scale reduction factor” \hat{R} for each parameter, which is defined to be the ratio of overall pooled variance (*i.e.* calculated using samples taken from all runs combined) to the mean variance within each run. Since MCMC is initialised randomly in parameters space, \hat{R} starts large and is expected to approach one in the limit of large iteration number. Values less than 1.1 are considered to be indicative of convergence. If, however, \hat{R} does not reach this threshold (even after a very large number of iterations) points to the existence of multimodality. Under these circumstances the results from BICl cannot be trusted. Note, \hat{R} relies on comparing independent MCMC runs, and so cannot be calculated for a single run.

4.3.9 Population plots

This page shows how posterior estimates for populations dynamically change as a function of time. These variations can be viewed in two different ways (selected from the drop-down menu in Fig. 33A):

As a graph – Figure 33B shows an example taken from EX 8. Dynamic changes in the posterior population sizes in the model compartments are plotted as a function of time. The shaded areas denote 95% credible intervals. The observed classification can be changed (Fig. 33C) and individual compartments within this can be turned on or off by clicking on the relevant checkboxes. For models with more than one classification, further filters can be applied to focus on only those individuals in specified states (*e.g.* at a particular location).

As an animation – Figure 34A shows the spatial spread of individuals from the posterior at a particular point in time (taken from example EX 30). The model is shown in greyscale to represent population sizes in each of the compartments (with darker colours representing larger population size). Pressing the play button (Fig. 34B) starts an animation which shows the dynamic variation in these populations as a function of time. The animation can be paused, restarted or a particular time point can be viewed by clicking on the time bar. Several options can be selected regarding how the data is displayed: All the posterior samples can be used together or a specific one chosen (Fig. 34C). The colour scheme from the model or greyscale (Fig. 34D) can be used to represent population sizes (the ‘rescale’ option rescales the palette each frame to help highlight variation which would otherwise be difficult to see). The label on the compartments can be turned on or off and selected to include the compartmental name or not (Fig. 34E).

⁷ Here “mode” refers to a region of high posterior probability that MCMC locally optimises, but which might not represent the global optimum.

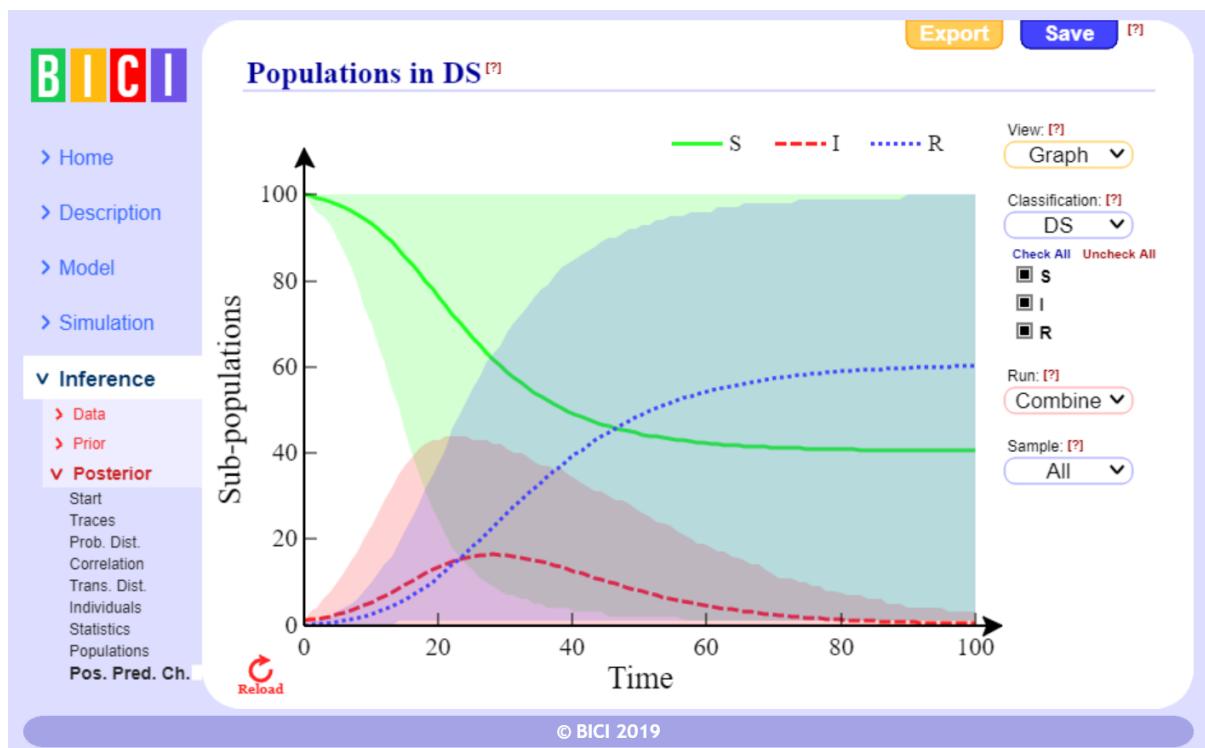


Figure 35 – Posterior predictive check: graph view. Shows the results of simulating using posterior parameter samples and initial states (EX 1). Lines give average behaviour and shaded areas encompass 95% of simulated results.

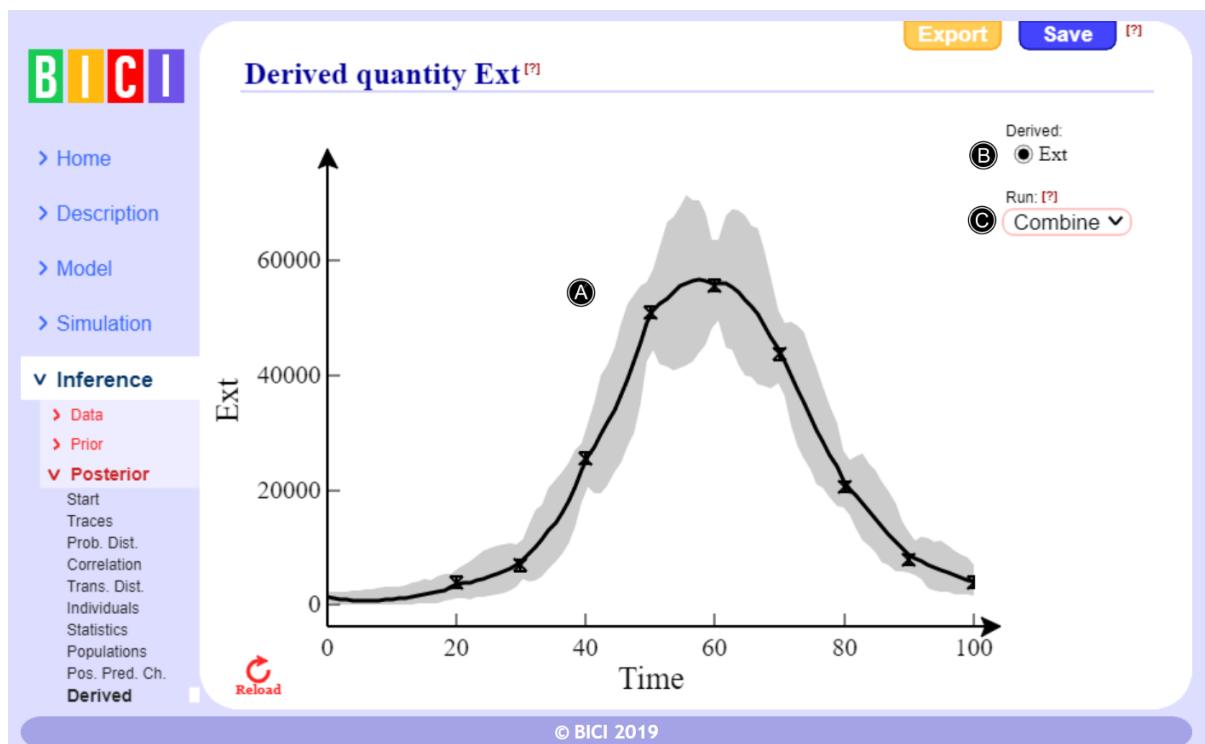


Figure 36 – Derived quantities. A: Shows how derived quantities change as a function of time (this example gives the environmental pathogen level in EX 10), B: Select quantity, C: Select an MCMC run or combine results from all runs.

4.3.10 Posterior predictive population plots

This shows how populations dynamically change as a function of time when the model is simulated using posterior samples for the parameters and initial conditions (hence generating a posterior predictive check). As for populations, the results can either be viewed either as a graph (where lines represent average behaviour and the shaded areas denote regions encompassing 95% of the simulations) or, as with Fig. 34, through an animation across the model itself (which shows population variation in greyscale). The results can be filtered by classification, MCMC run, or sample number.

Figure 35 shows an example of a graph for the SIR model taken from EX 1. The shaded regions in this particular case are large because the probability of epidemic extinction is high. This leads to two contrasting behaviours in the simulations: one in which epidemics occur (similar to that shown in Fig. 9) and one in which they do not (the initially infected individual recovers, leaving almost all individuals susceptible for the entire time period)⁸.

4.3.11 Derived

This section is only displayed if there exists time varying derived quantities in the model (see §2.5 for details). Figure 36A shows an example of a selected derived quantity dynamically changing as a function of time. Lines represent posterior means and the shaded areas give 95% credible intervals. Measured data (if it exists) is plotted as crosses with associated error bars. Different derived quantities can be selected in Fig. 36B and results can be filtered by MCMC run in Fig. 36C.

5 Input and output

5.1 Loading and saving

BICI permits users to load and save analyses in a special “.bici” format (see Fig. 1B). This is useful because it conveniently allows description, data, and analysis to all be contained in a single file for future reference. Also those publishing results using BICI can include the “.bici” file in the supplementary material such that analysis can transparently be reproduced by readers of the paper. When saving, two options are available: “With results” includes posterior samples along with the model and data (so that inference does not need to be run again when the file is loaded), and “W/o results” which does not store the posterior samples (leading to a much smaller file size that can, for example, be emailed).

⁸ Combining these two drastically different possibilities leads to the large posterior predictive uncertainty in Fig. 35.

5.2 Exporting

Outputs from BICI can be achieved by clicking on the “Export” button on the top right hand corner (*e.g.* see Fig. 24D). A number of exporting possibilities exist:

- **Print** – For many pages this option allows the user to print the content on the current page (*e.g.* the model and most of the simulation and posterior visualisation options).
- **Graph (.png)** – Instead of printing the output is turned into an image that can be saved.
- **Graph (.txt)** – This outputs the raw data as a text file, so they can be loaded into other software.
- **Parameters** – This outputs posterior parameter samples as a tab separated table in text format (for subsequent analysis in other software, specifically they can be loaded into the well-known Tracer software).
- **Events** – This outputs posterior event samples in text format (for subsequent analysis in other software). Specifically, a tab separated table is generated with four columns: 1) Sample number, 2) individual ID, 3) birth time, 4) transitions for that individual.
- **Diagnostics** – This provides information regarding the performance of the underlying MCMC proposals.

```
description text="

This example is taken from epidemiology.
Individuals are classified as susceptible 'S', infected 'I' or recovered 'R'.
"

# Defines the model
compartment classification="DS" name = "S" x="0" y="1" color="#00ff00"
compartment classification="DS" name = "I" x="1" y="1" color="#ff0000"
compartment classification="DS" name = "R" x="2" y="1" color="#0000ff"
transition from="S" to="I" type='exponential' rate='[β]*{I}'
transition from="I" to="R" type='exponential' rate='[γ]'

# Defines parameters used for simulation
siminitpopulation S="100" I="1" R="0"
setparam param="β" value="0.003"
setparam param="γ" value="0.1"

# Defines priors
setprior param="β" prior="flat" min="0" max="1"
setprior param="γ" prior="flat" min="0" max="1"

# The data
data type="transition" name="S->I" from='S' to="I" min="0" max="100" table="
Ind. 2 18.3949
Ind. 3 30.0119
:
"
data type="transition" name="I->R" from='I' to="R" min="0" max="100" table="
Ind. 1 24.7636
Ind. 2 25.8068
:
"
"
```

Figure 37 – Importable file. This shows an example of an importable text file that is used to define the model, priors and data for example EX 1. The ‘:’ symbols denote lists of data that have been omitted for brevity. The full file is available at ‘Models/Model and data for EX1.txt’.

5.3 Importing

Constructing complicated models using the point and click interface can be time consuming and cumbersome. In these instances BICI allows the model and/or prior and/or data to be imported as a script in a '.txt' text file.

An example of this, albeit a simple one, is shown in Fig. 37 (this can actually be used to generate example EX 1 in §6). Each line consist of a command followed by a number of different options specifying how the command is implemented. For example, the line:

```
compartment classification="DS" name = "S" x="0" y="1" color="#00ff00'
```

implements the command 'compartment', which adds a compartment named 'S' to the 'DS' classification, gives it the colour green (note, colours are defined using RGB hexadecimal format) and places it at the location (0,1) on the model page. A list of all BICI commands along with their options is given in Appendix A. Note, the symbol '#' is used to add comments to the script, as seen in Fig. 37.

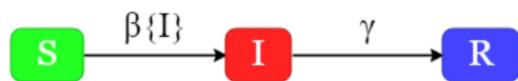
6 Examples

Example applications can be selected on the main page (see Fig. 1D). These demonstrate the versatility of BICI applied to a wide variety of different model and data scenarios. They can be altered or experimented on in any way, and new users are encouraged to try different possibilities as a way to familiarise themselves with the BICI interface (note, reloading examples from the main page returns them to their default settings).

The datasets referred to below (available in the "Datasets" folder of BICI) have been generated by simulating from the corresponding models and selecting the 'Simulation→Run→Gen. Data' option to generate tab separated text tables which represent potential real-world data collection scenarios. Using these to recreate the analysis in the examples is a useful way to learn how analysis in BICI is performed.

6.1 SIR model

This simple model is used to describe epidemic behaviour. Individuals are classified as being either susceptible to infection 'S', infected and infectious 'I', or recovered/removed/dead 'R'. A susceptible individual has a probability per unit time of becoming infected (referred to as the force of infection) as a result of other infected individuals (given by transmission rate parameter β multiplied by the total number of infected individuals $|I|$). For those individuals that do become infected, a recovery rate γ determines how quickly they recover. The following examples investigate inference under different data scenarios:



EX 1: Complete knowledge of events – Simulated data consists of the infection (Dataset 1) and recovery (Dataset 2) times of all individuals in the system, along with their initial disease status (Dataset 3). This represents complete knowledge of the underlying transition dynamics ξ . Although unlikely in a real-world setting, this nevertheless provides a best case scenario that represents maximum posterior precision for a given system. Simulation was carried out with 100 initially susceptible individuals and a single infected ($\beta=0.003$ and $\gamma=0.1$).

Running inference (by going to the ‘Inference→Posterior→Run’ page and clicking on the ‘Start’ button) and viewing the posterior probability distributions (on the ‘Prob. Dist.’ tab) shows that BICI is able to reasonably accurately estimate the parameter values used to generate the data (*i.e.* by selecting the model variables β and γ in the top right-hand corner, we find the posterior distributions are highly peaked near to $\beta=0.003$ and $\gamma=0.1$).

The basic reproductive ratio R_0 is defined as the number of infections caused by an infected individual in an otherwise susceptible population. This is an important number because epidemics proliferate or die out depending on whether R_0 is greater or less than one. R_0 can be derived from the model parameters as $R_0=100\times\beta/\gamma$ (see the ‘Model→Derived’ tab to see how this is incorporated into the model). Posterior distributions show that R_0 is peaked around its true value of $100\times0.003/0.1=3$.

The file ‘Model and data for EX1.txt’ in the ‘Models’ directory shows an importable script which can be used to define the model, prior and data in this particular example. Whilst not strictly necessary for such a simple model (with the point and click interface sufficient), this approach becomes essential for complex models, *e.g.* spatial (see §5.3 for details).

EX 2: Known initial state and recoveries – This illustrates an example in which infection times are unknown (this is common in many experimental and real-world applications, especially when the “recovery” state corresponds to the death of individuals). Simulated data consists of the recovery times (Dataset 2) and initial disease status (Dataset 1) of 100 initially susceptible individuals and a single infected ($\beta=0.003$, $\gamma=0.1$).

EX 3: Known recoveries only – Here the data for inference consists of just the recovery times of individuals. A small external force of infection is assumed to initiate an epidemic at some unknown point in time (this is incorporated into the model by an additional 0.0001 added to the force of infection $\beta\{I\}$). Simulated data consists of the recovery times (Dataset 4) of 100 initially susceptible individuals ($\beta=0.003$ and $\gamma=0.1$). Note, on the ‘Inference→Data→Sources’ page the data source named “Init.” lists all individuals in the system and defines them to be in the susceptible ‘S’ state prior to the beginning of the epidemic (Dataset 5).

EX 4: Periodic disease status – One strategy to monitor if and when an epidemic occurs is by periodically testing individuals to ascertain their disease status. Simulated data consists of disease status measurements taken on every individual from time $t=0$ to $t=60$ in steps of 10 time units (Dataset 6), which was generated with 100 initially susceptible individuals and a single infected ($\beta=0.003$ and $\gamma=0.1$).

EX 5: Disease diagnostic test results – In many real-world settings it is impossible to precisely ascertain disease status. Rather a disease diagnostic test is carried out which gives result 1/0 indicating positive/negative to being in the infected ‘I’ state. Inaccuracy in the test is characterised by the sensitivity Se (probability a truly infected individual tests positive) and specificity Sp (probability a truly uninfected individual tests negative, see §4.1.1 for further details). In this example simulated data consists of periodically measured disease diagnostic test results (Dataset 7), with known sensitivity $Se=0.5$ and specificity $Sp=0.95$, for 100 initially susceptible individuals and a single infected ($\beta=0.003$ and $\gamma=0.1$). The initial disease status of individuals is assumed to be known (Dataset 3).

EX 6: Future and past prediction – This example illustrates the ability of BICI to not only predict future dynamic behaviour but also predict behaviour prior to the time from which data has been collected.

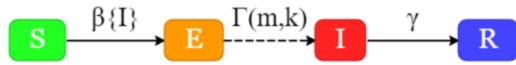
A small external force of infection is assumed to initiate an epidemic at some unknown point in time (this is incorporated into the model by an additional 0.0001 added to the force of infection $\beta\{I\}$). Simulated data consists of disease status measurements made at times $t=80, 100$ and 120 , during an epidemic (Dataset 8) for 100 initially susceptible individuals ($\beta=0.003$ and $\gamma=0.1$). The behaviour before and after these times is inferred. Before the epidemic all individuals are assumed to be susceptible, which is incorporated into the model by setting ζ_s to be large in the Dirichlet prior on the initial state (see the ‘Inference→Prior’ page).

EX 7: Time classification – This example illustrates how changes in model parameters over time can be incorporated into the model. The rate at which infections and recoveries occur is characterised by the parameters β_{Time} and γ_{Time} (note, they have the special subscript ‘Time’). The ‘Model→Time’ tab is defined to have two compartments, ‘ $T<40$ ’ and ‘ $T>40$ ’, which relate to times prior to or after $t=40$. In other words at $t=40$ some change occurs in the system (e.g. a change in management conditions, a medical intervention, or an alteration in the propensity for individuals to come into contact), which results in the rate parameters changing. Because this change is sudden, a time smoothing prior for β_{Time} and γ_{Time} is not used.

Simulated data consists of the infection (Dataset 9) and recovery (Dataset 10) times and initial infection status (Dataset 3) of 100 initially susceptible individuals and a single infected ($\beta_{T<40}=0.002$, $\gamma_{T<40}=0.08$, $\beta_{T>40}=0.004$ and $\gamma_{T>40}=0.12$).

6.2 The SEIR model

This simple model is used to describe epidemic behaviour. Individuals are classified as being either susceptible to infection ‘S’, exposed (which means ‘infected but not infectious’) ‘E’, infectious ‘I’, or recovered/removed/dead ‘R’. A susceptible individual has a probability per unit time of becoming infected as a result of other infected individuals given by β times the number of infected individuals $\{I\}$. For those individuals that do become exposed, a gamma distributed incubation period (with mean m and shape parameter k) determines when they become infectious and a recovery rate γ determines how fast they recover.



EX 8: Periodic disease status – Simulated data consists of periodic disease status measurements (e.g. in livestock these can be established by combining blood tests for disease presence with nasal swab measurements for infectivity) made every 30 time units (Dataset 11) for 100 initially susceptible individuals and a single infected ($\beta=0.003$, $m=10$, $k=3$ and $\gamma=0.1$).

EX 9: Disease diagnostic test results – Simulated data consists of results from two disease diagnostic tests made on every individual every 20 time units. The first is sensitive to the ‘E’ and ‘I’ states (Dataset 12) (e.g. typical of a blood test), and the second is sensitive to only the ‘I’ state (Dataset 13) (e.g. typical of a nasal swab measurement). The initial disease status of individuals is assumed known (Dataset 14). Simulation were carried out with 200 initially susceptible individuals and a single infected ($\beta=0.003$, $m=10$, $k=3$, $\gamma=0.1$, $Se_{\text{Blood}}=0.7$, $Sp_{\text{Blood}}=0.98$, $Se_{\text{Nasal}}=0.5$ and $Sp_{\text{Nasal}}=0.95$).

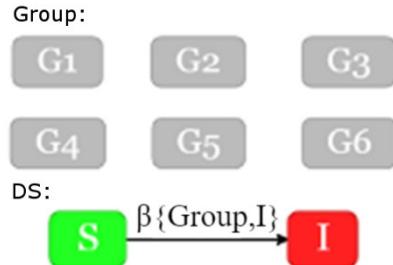
EX 10: Environmental and diagnostic test results – This provides an example in which the measurements made on the system are not quantities in the model itself, but are functionally dependent on these quantities. Here the relationship between the number of infected individuals and environmental pathogen level is assumed to be a simple linear relationship (which neglects accumulation of pathogen in the environment), with the proportionality constant estimated from the data. This dependency is incorporated into the model on the "Model→Derived" page.

Simulated data consists of periodic disease diagnostic test results made on the system (every 20 time units) which are sensitive to the 'E' and 'I' states (Dataset 15) (e.g. typical of a blood test) as well as measurements of environmental contamination (Dataset 16). The initial disease status is assumed to be known (Dataset 14). Simulations were carried out with 200 initially susceptible individuals and a single infected ($\beta=0.003$, $m=10$, $k=3$, $\gamma=0.1$, $Se_{Blood}=0.5$ and $Sp_{Blood}=0.95$).

EX 11: Estimating test sensitivity and specificity – This example shows that results from multiple, independent disease diagnostic tests can be combined to provide mutual estimates for each of their sensitivities and specificities. Simulated data consists of three sets of periodic disease diagnostic test results made on individuals (every 20 time units). The first (Dataset 17) is sensitive to the 'I' state and is representative of a culture test (low sensitivity, high specificity). The other two tests are sensitive to the 'E' and 'I' states (Datasets 18 and 19) and are representative of blood tests. The initial disease status is assumed to be known (Dataset 14). Simulation was carried out with 200 initially susceptible individuals and a single infected ($\beta=0.003$, $m=10$, $k=3$, $\gamma=0.1$, $Se_{Cult}=0.4$, $Sp_{Cult}=0.999$, $Se_{Blood1}=0.8$, $Sp_{Blood1}=0.95$, $Se_{Blood2}=0.7$ and $Sp_{Blood2}=0.98$).

6.3 Disease transmission experiments

Disease transmission experiments are used to discover the rate at which individuals become infected and to investigate factors affecting this rate (e.g. the role of fixed effects such as vaccination status). Typically populations of initially infected and uninfected individuals are placed in closed 'contact groups'. The simple example model here has two classifications: 'Group', which contains six contact groups, and 'DS', which gives the disease status of individuals within these groups. Susceptible individuals become infected at a rate given by β times the number of infected individuals sharing the same epidemic group {Group,I}. Data is collected on the ensuing epidemics (in a variety of ways) to estimate model parameters.



EX 12: Known infection times and initial state – Simulated data consists of the infection times of individuals (Dataset 20) along with their disease status and epidemic group at the beginning of the experiment (Dataset 21). Each epidemic group contains 20 individuals with one initially infected ($\beta=0.005$).

EX 13: Disease diagnostic test results – Simulated data consists of periodic disease diagnostic test results (Dataset 22) along with the disease status and epidemic group for each individual at the beginning of the experiment (Dataset 21). Each epidemic group contains 20 individuals with one initially infected ($\beta=0.005$, $Se_{Blood}=0.5$ and $Sp_{Blood}=0.95$).

EX 14: Initial / final disease status – Simulated data consists of the initial (Dataset 21) and final (Dataset 23) disease statuses of individuals along with the epidemic group they belong to (Dataset 21). Each epidemic groups contains 20 individuals with one initially infected ($\beta=0.005$).

EX 15: Incorporating vaccination status – This example illustrates incorporation of a fixed effect, in this case the effect of vaccination on disease transmission. The model has a third classification “Vac”, which contains the compartments ‘V’ for vaccinated and ‘NV’ for not vaccinated. Simulated data consists of periodic disease diagnostic test results taken on each individual (Dataset 24) along with the epidemic group they belong to, their vaccination status, and initial disease status (Dataset 25). Each epidemic group contains 20 individuals with one initially infected ($\beta_V = 0.003$, $\beta_{NV} = 0.006$, $Se_{Blood}=0.5$ and $Sp_{Blood}=0.95$).

EX 16: Incorporating group effect – This example illustrates how distributions can be imposed onto model parameters. In this case the force of infection is given by $\beta\{Group, I\}e^{\alpha_{Group}}$, where α_{Group} are group specific parameters that allow for variation in the overall speed of epidemics in different groups (e.g. due to different environmental conditions). This variation is assumed to be drawn from a normal distribution with mean zero and standard deviation σ (as is done for a random effect). This is incorporated into the model on the ‘Model→Distributions’ page. Simulated data consists of periodic disease diagnostic test results (Dataset 26) along with the disease status and epidemic group for each individual at the beginning of the experiment (Dataset 27). Here there were 12 epidemic groups each containing 20 individuals with one initially infected ($\beta=0.005$, $\sigma=0.5$, $Se_{Blood}=0.5$ and $Sp_{Blood}=0.95$).

EX 17: SNP effects on susceptibility / infectivity – This scenario considers a disease transmission experiment in which the genetic status at a given locus (represented by a single nucleotide polymorphism or 'SNP') affects both the susceptibility and infectivity of an individual. Assuming the SNP has variants ‘A’ and ‘B’ and individuals are diploid then three possible genotypes exist: ‘AA’, ‘AB’ and ‘BB’. The model has a third classification ‘SNP’ which contains a compartment for each of these possibilities. The force of infection is given by $\beta_{SNP}(f_{AA}\{Group, I, AA\} + \{Group, I, AB\} + f_{BB}\{Group, I, BB\})$, such that β_{AA} , β_{AB} and β_{BB} measure the susceptibility of the three genotypes and f_{AA} and f_{BB} give the relative infectivity of genotypes AA and BB, respectively, compared to AB. Simulated data consists of periodic disease diagnostic test results (Dataset 28) along with the disease status, epidemic group and SNP value for each individual at the beginning of the experiment (Dataset 29). Simulations were carried out with 64 epidemic groups each containing 10 individuals with one initially infected ($\beta_{AA}=0.006$, $\beta_{AB}=0.01$, $\beta_{BB}=0.015$, $f_{AA}=0.5$, $f_{BB}=1.5$, $Se_{Blood}=0.5$ and $Sp_{Blood}=0.95$).

EX 18: SIR with non-Markovian recovery – This example considers a disease transmission experiment with individuals exhibiting more complicated disease dynamics. Specifically an SIR model with a gamma distributed recovery duration is assumed (with mean m and shape parameter k). Simulated data consists of periodic disease diagnostic test results (Dataset 30) along with the disease status and epidemic group for each individual at the beginning of the experiment (Dataset 31). Simulations were carried out with 6 epidemic groups each containing 20 individuals, one of which was initially infected ($\beta=0.01$, $m=20$, $k=3$, $Se_{Blood}=0.5$ and $Sp_{Blood}=0.95$).

6.4 Population with births and deaths

This simple model can be used to capture stochastic variation in population size in a wildlife setting. The rate at which individuals are born is given by $v\{P\}(1 - \{P\}/\kappa)$, where v is the birth rate, $\{P\}$ is the current population size and κ is the carrying capacity (introduced to limit the maximum population size as a result of locally available resources, e.g. food and space). Individuals die with constant mortality rate μ .



EX 19: Periodic population estimates – Simulated data consists of periodic population estimates (Dataset 32) taken every 10 time units with a single initial individual ($v=0.4$, $\kappa=100$ and $\mu=0.1$).

EX 20: Death times – Simulated data consists of the death times of individuals (Dataset 33) along with the initial population (Dataset 34), which contains a single individual ($v=0.4$, $\kappa=100$ and $\mu=0.1$).

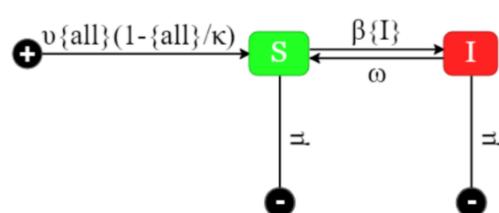
EX 21: Captures – This example considers the case of a capture-mark-recapture trial. Simulated data consists of periodic capture campaigns with capture probability p (Dataset 34 gives which individuals were caught at which times, and Dataset 35 gives when the campaigns were carried out) ($v=0.4$, $\kappa=100$, $\mu=0.1$ and $p=0.4$). A single initial individual is assumed. Note, because individuals can be born and die without ever being observed, this analysis uses the "Unobserved individuals" option on the 'Inference→Data→Sources' page.

EX 22: Age dependent mortality – In real wildlife populations the mortality rate may not be constant during an individual's lifetime. This example uses capture-mark-recapture data to estimate this variation in mortality. Individuals are classified into 6 age categories (as seen on the 'Model→Age' page). Simulated data consists of periodic capture campaigns with capture probability p (Dataset 37 gives which individuals were caught at which times, and Dataset 38 gives when the campaigns were carried out). Simulations were carried out with initial population size 100 ($v=0.4$, $\kappa=200$, $\mu_{A0-0.01}=0.4$, $\mu_{A0.01-2}=0.3$, $\mu_{A2-4}=0.2$, $\mu_{A4-6}=0.09$, $\mu_{A6-8}=0.07$, $\mu_{A8-10}=0.1$, $\mu_{A10+}=0.2$ and $p=0.4$).

EX 23: Captures and a fraction of death times – Individuals are captured at periodic time points (with capture probability p) and this provides data to inform inference. A fraction of death events (which can be estimated) are also observed to provide additional information. Simulated data consists of periodic capture campaigns (Dataset 39 gives which individuals were caught at which times, and Dataset 40 gives when the campaigns were carried out) as well as observing individual deaths (Dataset 41) with probability f . Simulations were carried out with initial population size 200 ($v=0.3$, $\kappa=400$, $\mu=0.1$, $p=0.4$ and $f=0.6$).

6.5 SIS model with births and deaths

This model not only captures stochastic variation in population size in a wildlife setting, but also accounts for changes in individual disease status. As well as the usual infection transition from 'S' to 'I' with rate $\beta\{I\}$, this model includes a reverse transition with rate ω to account for the fact that individuals do not gain lifelong immunity (true for some diseases such as influenza). Individuals are born at rate $v\{all\}(1 - \{all\}/\kappa)$, where v is the birth rate parameter, $\{all\}$ (which alternatively can be written $\{\}$)



is the current population size and κ is the carrying capacity (introduced to limit the maximum population size as a result of locally available resources, e.g. food and space). Individuals die with mortality rate μ .

EX 24: Periodic disease status measurements – Simulated data consists of the disease status of every individual (i.e. the probability of capture is $p=1$) every 20 time units (Dataset 42) along with capture data (Dataset 43) which provides the timings of these measurements (note, this is necessary because it ensures that only observed individuals are alive at these times). Simulations were carried out starting with 150 susceptible and 100 infected individuals ($\beta=0.0005$, $\omega=0.05$, $v=0.1$, $\kappa=300$ and $\mu=0.02$).

EX 25: Disease status from captures – Simulated data consists of the disease status of captured individuals (Dataset 44), as well as the timings of capture campaigns (Dataset 45), which were carried out every 5 time units with capture probability $p=0.4$. Simulations started with 100 susceptible and 100 infected individuals ($\beta=0.001$, $\omega=0.05$, $v=0.1$, $\kappa=300$ and $\mu=0.05$).

EX 26: Disease diagnostic test results from captures – Simulated data consists of disease diagnostic test results from captured individuals (Dataset 46), as well as the timings of capture campaigns (Dataset 47), which were carried out every 5 time units with capture probability $p=0.4$. Simulations started with 100 susceptible and 100 infected individuals ($\beta=0.0015$, $\omega=0.05$, $v=0.1$, $\kappa=300$, $\mu=0.05$, $Se_{Blood}=0.5$ and $Sp_{Blood}=0.95$).

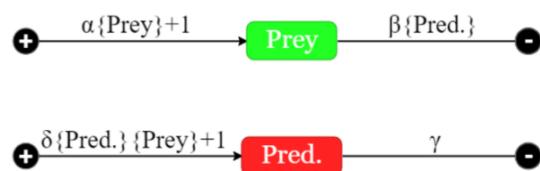
EX 27: Disease induced mortality – This shows an example in which infected individuals are assumed to be $m=3$ times more likely to die than uninfected individuals, a factor estimated from data. This disease-induced mortality is incorporated into the model by defining a mortality rate $m\mu$ for infected individuals. Simulated data consists of disease diagnostic test results from captured individuals (Dataset 48), as well as the timings of capture campaigns (Dataset 49), which were carried out every 5 time units with capture probability $p=0.4$. Simulations started with 100 susceptible and 100 infected individuals ($\beta=0.001$, $\omega=0.05$, $v=0.2$, $\kappa=300$, $\mu=0.05$, $m=3$, $Se_{Blood}=0.5$ and $Sp_{Blood}=0.95$).

6.6 Predator-prey model

These stochastic models predict dynamic, unstable variation in predator and prey populations.

Typically the food supply for prey (e.g. rabbits) is plentiful, so the rate at which they are born is given by the prey birth rate α times their population size $\{\text{Prey}\}$. Rather than being limited by local resources, the number of prey is regulated by being consumed by predators (e.g. foxes), so the mortality rate is given by β times the predator population size $\{\text{Pred.}\}$.

On the other hand, predators rely on prey for food, so the rate at which they are born is given by not only the predator birth rate δ times their population size $\{\text{Pred.}\}$, but also a further factor $\{\text{Pred}\}$. Predators have a constant mortality rate γ .



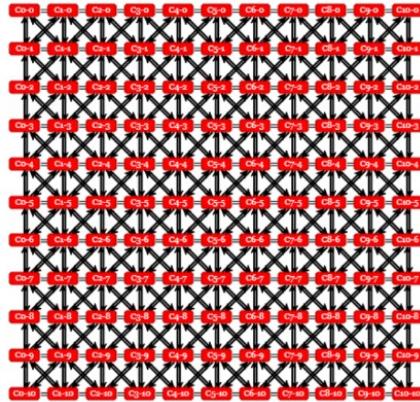
The description above represents the simple Lotka-Volterra model. On top of this a small influx of predators and prey (1 per unit time) is assumed to avoid stochastic population extinction.

EX 28: Population measurements – Simulated data consists of population estimates for the predators (Dataset 50) and prey (Dataset 51) every 2 time units. Simulations were carried out starting with 100 prey and 20 predators ($\alpha=0.666$, $\beta=0.01333$, $\delta=0.01$ and $\gamma=1$).

EX 29: Captures and death times – Simulated data consists of captured prey (Dataset 52) from capture campaigns carried out every time unit with capture probability $p=0.4$ (Dataset 53) as well as the death times of predators (Dataset 54). Simulations started with 100 prey and 20 predators ($\alpha=0.666$, $\beta=0.01333$, $\delta=0.01$ and $\gamma=1$).

6.7 Spatial diffusion model

This model captures the movement of individuals across a landscape. For simplicity, locations are divided into a 11×11 square grid of compartments. Transitions to neighbouring compartments occur at rate ω and diagonally at rate $\omega/2$. This represents a particularly simple case, but geographical features (such as boundaries, rivers, different types of terrain) could easily be incorporated by making changes to these transition weights.



EX 30: State measurements at all locations – Simulated data consists of location measurements of individuals at times $t=0$, $t=50$ and $t=100$ (Dataset 55) for 100 individuals initially started in the central 'C5-5' compartment ($\omega=0.02$). This represents an example in which creating the model using the point and click interface would be time consuming and difficult, so instead it has been imported from the file 'Model 1.txt' in the 'Model' directory.

EX 31: Captures at some locations – Simulated data consists of the captures (with probability $p=0.4$) of individuals at 9 different spatially separated locations (Dataset 56) periodically every 2 time units (Dataset 57). Simulations were carried out starting with 5 individuals in each location ($\omega=0.03$). The spatial model was imported from 'Models/Model 2.txt'.

6.8 Spatial disease model

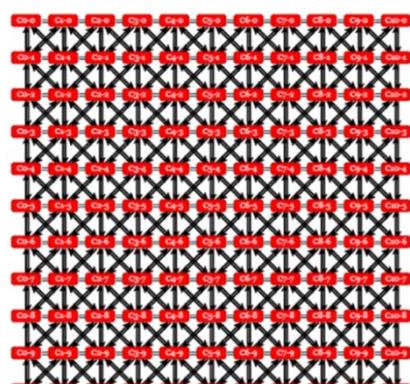
This simple model captures not only the movement of individuals across a landscape, but also their disease status.

The 'Loc' classification defines locations divided into a 11×11 square grid of compartments. Transitions to neighbouring compartments occur at rate ω and diagonally at rate $\omega/2$.

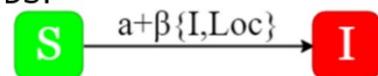
The 'DS' classification defines the disease status of individuals with 'S' and 'I' compartments representing susceptible and infected. The force of infection is given by β times the number of infected individuals sharing the same location $\{l, Loc\}$ plus a small external force of infection ($a=0.0001$).

EX 32: Location and disease status – Simulated data consists of periodic measurements of location and disease status every 20

Loc:



DS:



time units (Dataset 57). Simulations were carried out starting with 5 individuals in each location ($\omega=0.02$ and $\beta=0.2$). The spatial model was imported from ' Models/Model 1.txt'.

EX 33: Captures with disease diagnostic test results – Simulated data consists of measurements of location and disease diagnostic test results (Dataset 58) from captured individuals coming from capture campaigns (with capture probability p) every 20 time units (Dataset 59). Simulations were carried out starting with 5 individuals in each location ($\omega=0.02$, $\beta=0.2$, $p=0.4$, $Se_{Blood}=0.5$ and $Sp_{Blood}=0.95$). The spatial model was imported from ' Models/Model 1.txt'.

6.9 Location-based spatial disease model

This model is taken from epidemiology and describes the spatial spread of infection between different geographical locations. Rather than considering individuals in a population, here each location is fixed and has a disease status associated with it (S, I or R). This is setup in BICI by defining two classifications:

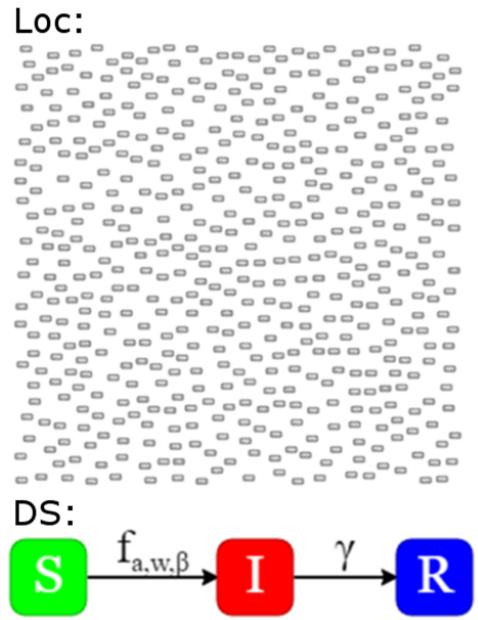
The 'Loc' classification contains compartment for different geographical locations, each containing a single individual representing that location (and so individuals do not move). In this example locations are randomly sampled within a square geographical region of size 100×100 units, with sampling restricted to ensure a minimum distance of 3.5 units between neighbours, e.g. to represent farms.

The 'DS' classification represents the disease status of locations. The force of infection on location l is given by $a + w \sum_q \{I, q\} K(d_{l,q})$ where a is a small external term (used to initiate epidemics), w determines the rate of spatial spread, the sum goes over all other locations q , $\{l, q\}$ takes the values 1 or 0 depending on whether location q is infected or not and $K(d_{l,q})$ is a spatial kernel which depends on the Euclidian distance $d_{l,q}$ between locations l and q . Infected locations recover at rate γ .

Two different types of spatial kernels are considered:

EX 34: Power-law spatial kernel – A power law kernel is assumed with a cut-off distance of 15 units: $K(d_{l,q}) = d_{l,q}^\beta$ for $d_{l,q} < 15$ else zero. Simulated data consists of the disease status of locations every 30 time units (Dataset 60) ($w=0.4$, $a=5 \times 10^{-5}$, $\beta=-1.5$ and $\gamma=0.03$). File 'Models/Model 3.txt' was imported to create this model.

EX 35: Exponential spatial kernel – An exponential kernel is assumed with a cut-off distance of 15 units: $K(d_{l,q}) = \exp(-d_{l,q}/d_0)$ for $d_{l,q} < 15$ else zero. Simulated data consists of the disease status of locations every 30 time units (Dataset 62) ($w=0.1$, $a=5 \times 10^{-5}$, $d_0=5$ and $\gamma=0.03$). The file 'Models/Model 4.txt' was imported to create this model.



6.10 Spatial disease model with births and deaths

This model is taken from epidemiology and describes the spatial spread of infection between different badger social groups. Badgers are a known carrier of bovine TB (bTB), so understanding its spread in the wild has important implications for the eradication of bTB in cattle populations. This model contains two classifications:

The ‘Loc’ classification defines the geographical locations of different badger social group. For simplicity badgers are assumed to only move between nearest neighbour social groups at a rate ω .

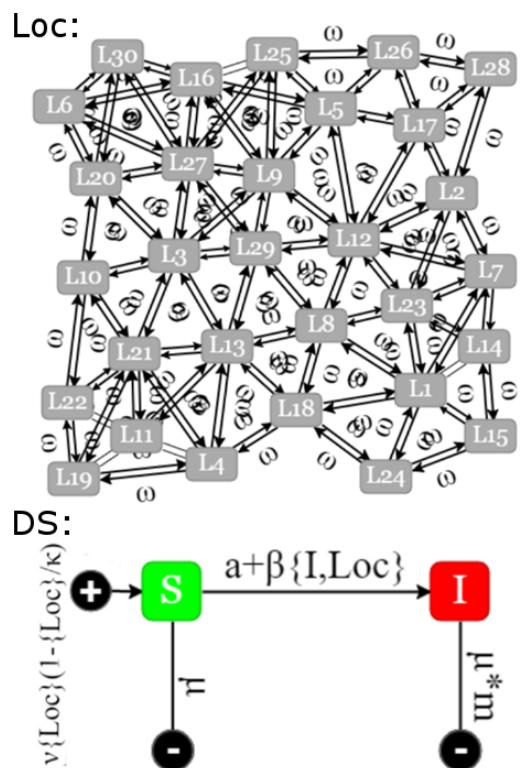
The ‘DS’ classification defines the disease status of badgers, represented by a simple SI model, along with demographics. Susceptible badgers become infected either from a small external force of infection (defined by a) or from infected badgers sharing the same social group (through transmission rate β). Demographics is incorporated through a birth rate v , a carrying capacity κ and mortality rate μ (which is multiplied by a factor m for infected badgers to incorporate disease induce mortality).

EX 36: A model of badger social groups – Simulated data consists of location and disease diagnostic test results (Dataset 63) from capture campaigns carried out every 2 time units (Dataset 64), with probability of capture $p=0.4$ ($\omega=0.01$, $v=0.2$, $\kappa=20$, $\mu=0.1$, $m=2$, $a=0.0001$, $\beta=0.1$, $Se_{Blood}=0.5$ and $Sp_{Blood}=0.95$). The file 'Models/Model 6.txt' was imported to create this model.

7 Code

The code for BICI is split into two parts:

- **The interface** – This is written in JavaScript and runs on the desktop by means of NW.js. For those interested, the code consists of the “index.html” file in the main directory⁹ and the supporting JavaScript files in the “js” directory.
- **The core code** – Performs the MCMC Bayesian analysis when BICI is executed. This is written in highly efficient C++ code which can be found in the “Execute” directory (this consists of the “BICI.cc” main file along with numerous header files).



⁹ On the Macintosh platform this is located in the “SIRE.app/Contents/Resources/app.nw/” folder.

8 License and warranty

BICI is free software under the terms of the GNU General Public License version 3 www.gnu.org/licenses/gpl-3.0.en.html. This allows users to redistribute and/or modify BICI. The program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY.

9 Citing BICI

We kindly request that those who do use BICI analysis in their publications cite this tool:

Pooley CM, Doeschl-Wilson AB, Marion G., *BICI – A flexible tool for simulation and inference from user-defined compartmental models*. To be submitted (2020)

10 Plans for BICI v2.0

One of the limiting factors in BICI v1.0 is that it only incorporates individual variation through categorisation. That is, two individuals behave in the same way if they share the same compartments in their classifications. However, often it is useful to include information that is not categorical. For example, covariates might be incorporated into rate expressions (e.g. mortality rate might realistically depend on an individual's weight). Furthermore, individual variation might be correlated between individuals, as is assumed in quantitative genetics models in which close family members are more likely to share similar trait values. The next version of BICI plans to incorporate these extensions.

Acknowledgments

BICI makes use of two other pieces of software and we would like to acknowledge their contribution. Firstly, NW.js (from the website nwjs.io/) was used to build the interface. Secondly, tinyXML (from the website www.grinninglizard.com/tinyxml/) was used by the C++ code to parse the XML file which provides initialisation information. Both of these software are excellent and highly recommended.

References

- [1] C. M. Pooley, S. C. Bishop, A. B. Doeschl-Wilson, and G. Marion, "Estimating genetic and non-genetic effects for host susceptibility, infectivity and recoverability using temporal epidemic data," *bioRxiv*, p. 618363, 2019.
- [2] E. Parzen, "On estimation of a probability density function and mode," *The annals of mathematical statistics*, vol. 33, no. 3, pp. 1065-1076, 1962.
- [3] R. E. Kass and A. E. Raftery, "Bayes factors," *Journal of the American Statistical Association*, vol. 90, no. 430, pp. 773-795, 1995.
- [4] H. Jeffreys, *The theory of probability*. OUP Oxford, 1998.
- [5] A. Gelman and D. B. Rubin, "Inference from iterative simulation using multiple sequences," *Statistical science*, vol. 7, no. 4, pp. 457-472, 1992.
- [6] B. Carpenter *et al.*, "Stan: A probabilistic programming language," *Journal of statistical software*, vol. 76, no. 1, 2017.

Appendix A

This provides a list of all the commands that can be used in an importable text file (see §5.3):

compartment	This adds a compartment to the model. <i>E.g.</i> compartment classification="DS" name="S" x="0" y="1" color="#ff0000"
	<i>classification</i> – The classification the compartment belongs to. <i>name</i> – The name of the compartment. <i>x,y</i> – The position on the screen in the model (rescaled to fit the screen). <i>Color</i> – The colour as specified in hexadecimal RGB format.
	Note, classifications themselves do not need to be specified as they are inferred from the compartment specifications.
transition	This adds a transition to the model. <i>E.g.</i> transition from="S" to="I" type="markovian" rate="[\beta]*{I}"
	<i>from</i> – The name of the compartment where individuals start. <i>to</i> – The name of the compartment when individuals end. <i>type</i> – Type of transition which can take one of the following possibilities: <ul style="list-style-type: none">• <i>markovian</i> – A “rate” property must be set that determines the probability per unit time of the transition occurring (alternatively a “time” can be set which is essentially the reciprocal of the rate).• <i>gamma</i> – The duration the individual stays in the “from” compartment before entering the “to” compartment is assumed to be gamma distributed. “mean” and “shape” properties must be set.• <i>weibull</i> – The duration the individual stays in the “from” compartment before entering the “to” compartment is assumed to be Weibull distributed. “lambda” and “k” properties must be set.• <i>source</i> – A “rate” property determines the rate at which individuals enter the system (note in this case no “from” property is set).• <i>sink</i> - A “rate” property determines the rate at which individuals leave the system (note in this case no “to” property is set). In cases in which filters are used, the expressions are put on different lines, <i>e.g.</i> To define a different infection rate for males and female then: transition from="S" to="I" type="markovian" rate="M:[\beta male]*{I} F:[\beta female]*{I} "
agettransitions	Sets the ages at which the compartments change in the “Age” classification. <i>E.g.</i> agetransitions values="0.5,1"
	<i>values</i> – The ages are comma separated.
timetransitions	Sets the times at which the compartments change in the “Time” classification. <i>E.g.</i> timetransitions values="50,100"
	<i>values</i> – The times are comma separated.
setparam	Sets a parameter value used for simulation. <i>E.g.</i> setparam param="beta" value="0.003"

	<p><i>param</i> – The name of the parameter used in the compartmental model.</p> <p><i>value</i> – The value used for simulation.</p>									
setprior	<p>Sets the prior for a model parameter.</p> <p><i>E.g.</i> setprior param="β" prior="flat" min="0" max="1"</p> <p><i>param</i> – The name of the parameter used in the compartmental model.</p> <p><i>prior</i> – The type of prior which can take one of the following possibilities:</p> <ul style="list-style-type: none"> • <i>flat</i> – A uniform prior probability within a range defined by the “<i>min</i>” and “<i>max</i>” parameters. • <i>gamma</i> – A gamma distribution with “<i>mean</i>” and standard deviation “<i>sd</i>” parameters. • <i>normal</i> – A normal distribution with “<i>mean</i>” and standard deviation “<i>sd</i>” parameters. • <i>log-normal</i> – A normal distribution with “<i>mean</i>” and standard deviation “<i>sd</i>” parameters (which are defined on the log scale). • <i>exponential</i> – An exponential distribution with a “<i>rate</i>”. • <i>beta</i> – A beta distribution with “<i>alpha</i>” and “<i>beta</i>” parameters. • <i>weibull</i> – A Weibull distribution with “<i>lambda</i>” and “<i>k</i>” parameters. • <i>fix</i> – Fixes a parameter to a particular “<i>value</i>”. 									
setdistribution	<p>This uses the same options as “<i>setprior</i>” except that here the distributions are incorporated as part of the model instead of the prior (hence when simulation is performed the values are sampled rather than set). See §2.6 for details.</p>									
setinitprior	<p>Sets the initial Dirichlet prior for the initial state of individuals.</p> <p><i>E.g.</i> setinitprior state="S,M" value="2"</p> <p><i>state</i> – Defines which states are referred to (for more than one classification they are comma separated).</p> <p><i>value</i> – The α value set in the Dirichlet prior.</p>									
addderived	<p>Adds a derived quantity to the model (see §2.5 for details).</p> <p><i>E.g.</i> addderived param="E" expression="[f]*{l}"</p> <p><i>param</i> – The name of the derived quantity.</p> <p><i>expression</i> – Equation relating the derived quantity to parameters and populations in the model.</p>									
siminitpop	<p>Defines initial population sizes for all compartments within a given classification (used for simulation initialisation).</p> <p><i>E.g.</i> siminitpopulation S="100" I="1" R="1"</p>									
siminitpercent	<p>Specifies the percentages of individuals in all but one compartment within a given classification (used for simulation initialisation for classifications other than the one specified in “<i>siminitpop</i>”). Note, the unspecified compartment is automatically set to ensure the percentages add to 100.</p> <p><i>E.g.</i> siminitpercent M="40"</p>									
initpopulation	<p>Defines the initial population.</p> <p><i>E.g.</i> initpopulation table="</p> <table style="margin-left: 20px; border-collapse: collapse;"> <tr> <td>Ind.1</td> <td>S</td> <td>M</td> </tr> <tr> <td>Ind.2</td> <td>I</td> <td>F</td> </tr> <tr> <td>"</td> <td></td> <td></td> </tr> </table> <p><i>table</i> – This specifies all the individuals initially present. The first column gives individual IDs. Subsequent columns (tab separated) give the states the individuals are initially in.</p>	Ind.1	S	M	Ind.2	I	F	"		
Ind.1	S	M								
Ind.2	I	F								
"										

simtimerange	The time range over which simulation is performed. <i>E.g.</i> simtimerange min="10" max="100"
	<i>min</i> – The starting simulation time. <i>max</i> – The ending simulation time.
simnumber	Determines the number of simulations to be performed. <i>E.g.</i> simnumber value= "10"
inftimerange	The time range over which inference is performed. <i>E.g.</i> inftimerange min="10" max="100"
	<i>min</i> – The starting inference time. <i>max</i> – The ending inference time.
runnumber	The number of independent MCMC runs in parallel. <i>E.g.</i> runnumber value= "3"
paramsampmax	The maximum number of parameter samples stored in memory (when this number is exceeded samples are thinned by a factor of two and gathered at half the rate). <i>E.g.</i> paramsamplemax value = "50000"
eventsampmax	The maximum number of event sequence samples stored in memory (when this number is exceeded samples are thinned by a factor of two and gathered at half the rate). <i>E.g.</i> eventsamplemax value = "10000"
termination	Determines when inference is stopped. <i>E.g.</i> termination type="converged" ess="500" R="1.2"
	<i>type</i> – This can take two possible values: <ul style="list-style-type: none"> • <i>none</i> – Inference is continued until manually stopped. • <i>converged</i> – Inference is stopped when the effective sample size for all the of the model parameters is above “ess” and the Gelman-Rubin statistic is below “R”.
description	Adds a description of the model/data/analysis (see §1.3). <i>E.g.</i> description text="This is a description of the model"
datadesc	Adds a description of the data (see §4.1). <i>E.g.</i> datadesc text="This gives a description of the data"
classdesc	Adds a description regarding a particular classification (see §2). <i>E.g.</i> classdesc classification="DS" text="This explains about a classification"
data (state)	Adds a state data source (see §4.1.1). <i>E.g.</i> data type="state" name="DataName" classification="DS" table="" Ind1 0 S : : : "
	<i>name</i> – Gives the name of the data (arbitrary, but useful for reference). <i>classification</i> – Defines which classification the data informs. <i>table</i> - The columns give: the individual ID, the time and the data value. The data value can take four different forms: <ul style="list-style-type: none"> • It is one of the compartment values (as in the example above). • It is from a list of possibilities and for each possibility a set of consistent states is defined, e.g. if “inf” and “notinf” are

measurements representing “infected” and “not infected”, then the properties *inf*=“I” *notinf*=“S,R” must be set.

- It is a diagnostic test result.
- It is from a list of possibilities and for each possibility an observation model is defined (which may depend on additional model parameters). For example assuming measurements ‘A’ and ‘B’, the observation model can be defined by

data type=“state” A=“S:[ω],I:1,R:[λ]” B=“S:1-[ω],I:0,R: 1-[λ]” ...

This means that if the individual is in state ‘S’ there is a probability ω of measuring ‘A’ otherwise ‘B’, if they are in the ‘I’ state ‘A’ is definitely measured, and if in the R state there is a probability λ of measuring ‘A’ otherwise ‘B’.

data (presence) Adds a presence data source (see §4.1.2).

E.g.

```
data type="presence" name="DataName" table=""
Ind1 0
:
"
```

name – Gives the name of the data (arbitrary, but useful for reference).

table - The columns give: the individual ID and the time observed.

data (transition) Adds a transition data source (see §4.1.3).

E.g.

```
data type="transition" name="DataName" from='S' to='I' Sex="M" min="10"
max="100" pd="on" detectprob="[pd]" table=""
Ind.1 20
:
"
```

type – The type of transition which can take one of the following possibilities:

- *transition* – gives information about changes of state within a classification.
- *source* – Individuals entering the system.
- *sink* – Individuals leaving the system.

name – Gives the name of the data (arbitrary, but useful for reference).

from – The name of the compartment where individuals start.

to – The name of the compartment where individuals end.

table – The columns give: the individual ID and the time of transition.

min – The minimum time from which transitions are observed.

max – The maximum time transitions are observed.

pd (optional) – If this is set to “on”, the “detectprob” equation defines the probability of observing the transition.

Note, other classifications can be specified (*e.g.* Sex=“M” in the example above means that only infection events on male individuals are observed).

data (move) Adds a move data source (see §4.1.4).

E.g.

```
data type="move" name="DataName" from='S' to='I' table=""
Ind.1 20
:
"
```

type – The type of data which can take one of the following possibilities:

- *move* – Externally imposes a change in state within a classification.
- *sourcemove* - Externally imposed individuals entering the system.
- *sinkmove* – Externally imposed individuals leaving the system.

name – Gives the name of the data (arbitrary, but useful for reference).

from – The name of the compartment where individuals start.

to – The name of the compartment where individuals end.

table – The columns give: the individual ID and the move time.

data (capture)	Adds a capture data source (see §4.1.5). <i>E.g.</i> data type="capture" name="DataName" pd="on" detectprob="[p]" table=" capname S,M 0 : : : "
	<i>name</i> – Gives the name of the data (arbitrary, but useful for reference). <i>pd</i> – The probability of detection can take two possibilities: <ul style="list-style-type: none"> • <i>off</i> – This means all individuals are assumed to be observed. • <i>on</i> – If the <i>detectprob</i> option is set, this gives the expected probability of observing an individual when a capture campaign takes place, otherwise a capturepd data source is expected to provide this information. <i>table</i> – The columns give: the name of the capture campaign, the compartments observed (if all compartments are observed then “All” is used and if multiple classifications are set, they are comma separated) and the time.
data (captureid)	Adds a captureid data source (see §4.1.6). <i>E.g.</i> data type="captureid" name="DataName" table=" Ind1 capname : : "
	<i>name</i> – Gives the name of the data (arbitrary, but useful for reference). <i>table</i> – The columns give: the individual ID and the name of the capture campaign.
data (capturepd)	Adds a capturepd data source (see §4.1.7). <i>E.g.</i> data type="capturepd" name="DataName" table=" capname [pp] : : "
	<i>name</i> – Gives the name of the data (arbitrary, but useful for reference). <i>table</i> – The columns give: the capture campaign name and an equation giving the probability of capture.
data (population)	Adds a population data source (see §4.1.8). <i>E.g.</i> data type="population" name="DataName" DS="I" table=" 4 100 20 5 120 20 : : : "

name – Gives the name of the data (arbitrary, but useful for reference).
table – The columns give: the estimate population size, the time of the measurement and the standard deviation in the estimate (assuming a gamma distributed variation about the mean).
Note, classifications can be defined such that only a subpopulation is assumed to be estimated, such as DS="I" is the example above.

data (derived)	Adds a derived data source (see §4.1.9). <i>E.g.</i> <code>data name="DataName" type="derived" quantity="E" table=" 4 100 20 : : : "</code>
	<i>name</i> – Gives the name of the data (arbitrary, but useful for reference). <i>quantity</i> – Defines which derived quantity is being measured. <i>table</i> – The columns give: the estimate derived quantity size, the time of the measurement and the standard deviation in the estimate (assuming a normally distributed error about the mean).
