MARCH 31, 2021

HEVTIMES APP

USER MANUAL

J. WELLS, S. BARRY & L. KELLY
UNIVERSITY OF STRATHCLYDE
Dept. Mathematics & Statistics, Glasgow, UK.,

INTRODUCTION

The HEVTimes project has developed a thermal death model for Hepatitis E Virus (HEV), to investigate the relationship between cooking times, temperatures, and the resulting reduction in virus particles when HEV is heated, within the context of food preparation.

The model uses existing data from the literature, alongside new data from experiments carried out at Glasgow Caledonian University, both of which describe the log-reduction in HEV over time, for various times and temperature combinations. A Bayesian approach is then used to fit a suitable predictive microbiology model to the data. Data is contained within a .csv type file, which can be easily updated to include new data, as it becomes available.

The model has been coded into a user-friendly Shiny app that runs within the RStudio software, and the following user manual is intended to guide the user through the operation of the resulting HEVTimes app. The user manual contains three main sections:

1. **QUICK START GUIDE**: To get up and running quickly.
2. **NOTES: USER MANUAL**: Some more in depth notes about the app processes.
3. **MATHEMATICAL MODEL**: Brief summary of the predictive microbiology model.
4. **UPDATING THE MAIN DATA FILE (HEVTIMES_DATA.CSV)**:
   Brief guide to using, editing, and updating with additional data.

QUICK START GUIDE:

1. Install **R software**: www.r-project.org . (Choosing correct version for your computer's operating system)

   Software download links can be found at cran.r-project.org/mirrors.html .

2. Install **RStudio Desktop software** (user interface for R): rstudio.com .

   Software download can be found at rstudio.com/products/rstudio/download/ .

3. Install **JAGS software** (Just Another Gibbs Solver mcmc-jags.sourceforge.net

   This handles the Bayesian simulations. Download at sourceforge.net/projects/mcmc-jags/files/ .

4. Install required **R Packages**:

   Simply open "HEVTimes_Install_Packages.R" in RStudio, highlight all (crtl+a) and hit "run" in top toolbar.

5. Provide **HEVTimes data** in appropriate .csv file format:

   This must be labelled "HEVTimes_Data.csv" and placed in the same folder as "HEVTimes_App.R" code.

6. Open "**HEVTimes_App.R**" in Rstudio. (Main code for the Shiny app):

   Hit "Run App" in the top toolbar. This should launch the app, as per Figure 1.

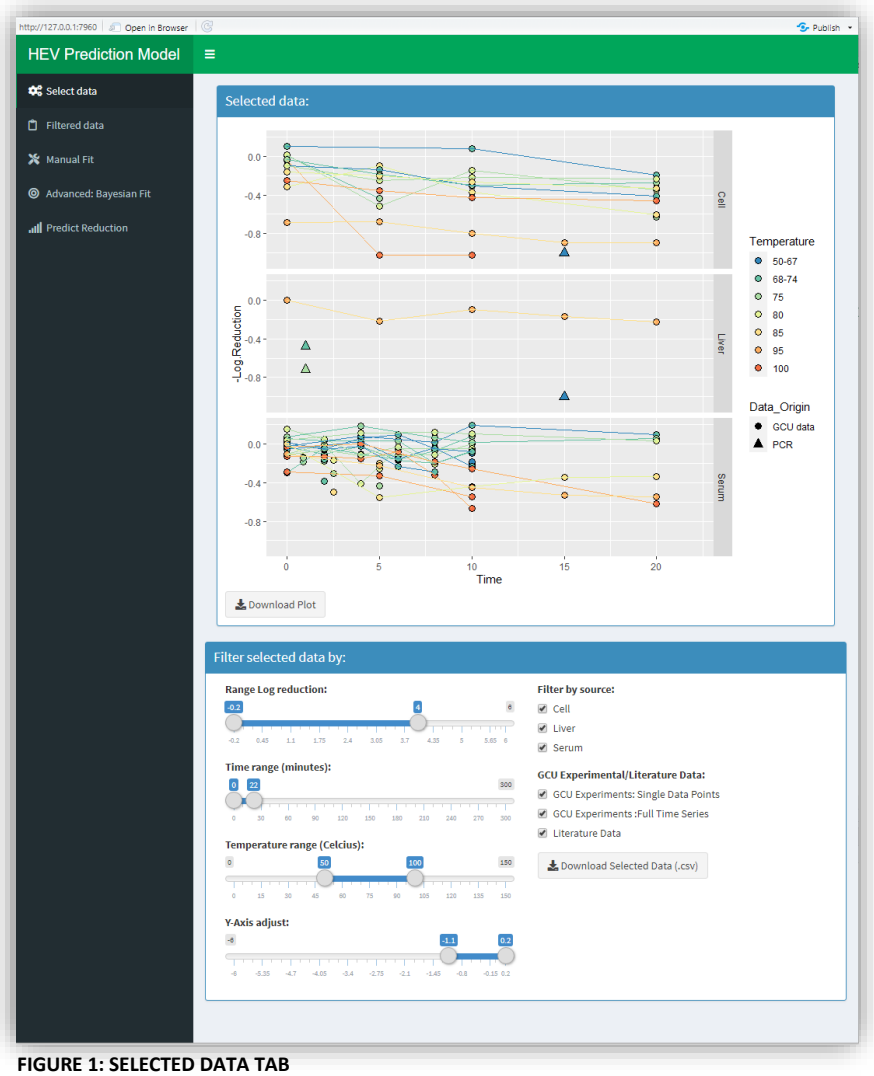

FIGURE 1: SELECTED DATA TAB

7. On the "**Select Data**" Tab (Figure 1):

   Use the sliders/check boxes to select the data you want to visualise/fit the model to. The figures will automatically adjust to show you the data you have chosen. Whilst the two "Download" buttons let you download this selected data as a .csv file or save the current plot.

   The data selected within this tab is then used by all the other tabs within the app.

8. Move to the "**Filtered Data**" Tab (Figure 2):

   This tab presents a searchable table, containing all of the data selected in the "Select Data" tab. Allowing you to see exactly which data points you are working with, and search/filter them by Temperature, Log Reduction, Time, etc. to gain further insights.

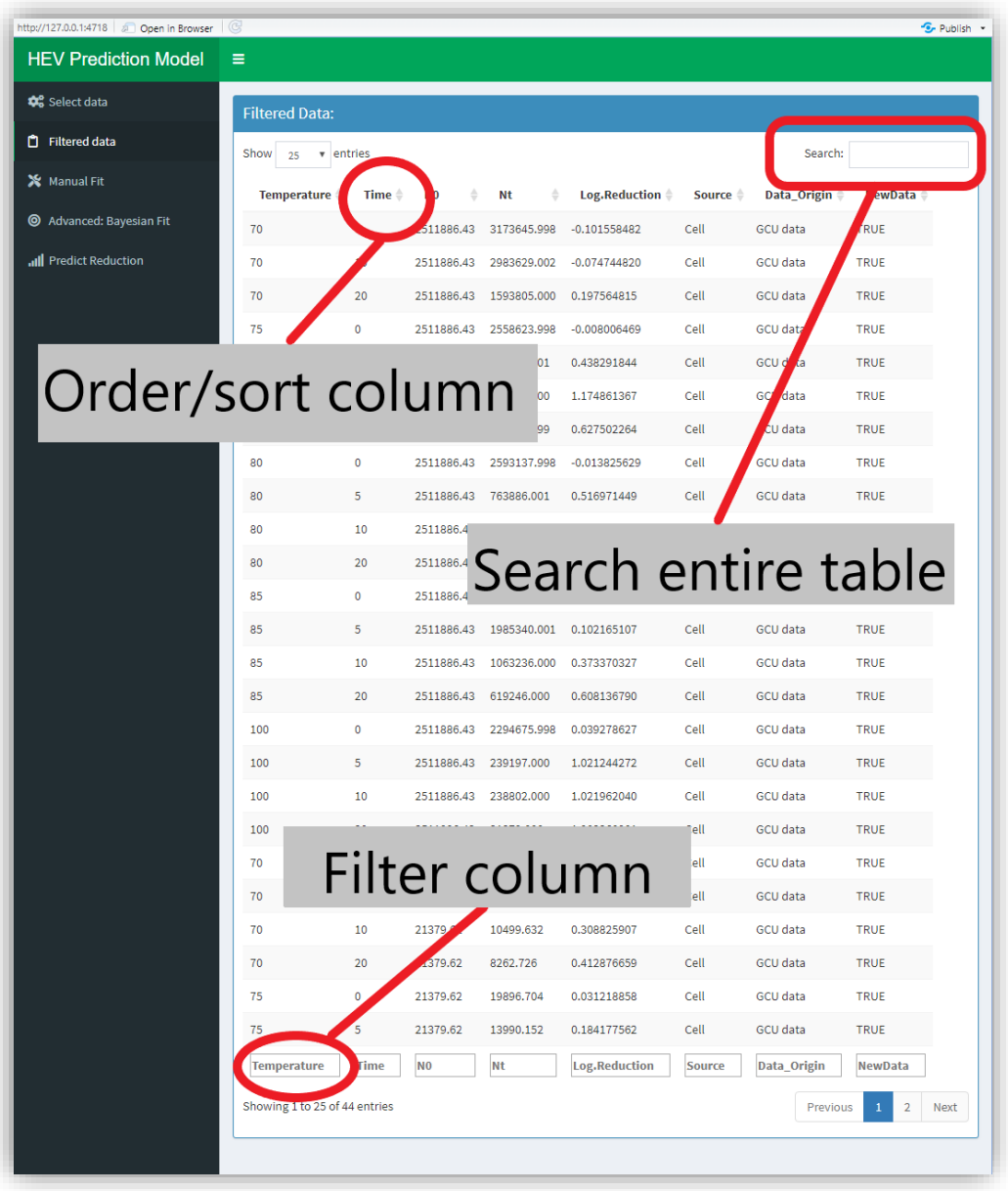

FIGURE 2: FILTERED DATA TAB

9. Move to the "**Manual Fit**" Tab (Figure 3):

[Note: This manual fit step is entirely optional. If desired, users can skip straight to Step 10.
This manual fit is intended as useful visual aid for both visualising the effect each model parameter has on the curve, as well as selecting suitable "initial guesses" for the true value of each parameter].

The model equation is presented at the top of this page, for reference.
Two sliders let you manually adjust the model parameters (A and β), to instantly see the effect they have upon the log-reduction curve produced by the model equation (Plotted in black).

Here, A is a constant frequency parameter and β is a constant shape parameter, and t**hese two parameters will be different for each temperature T .**

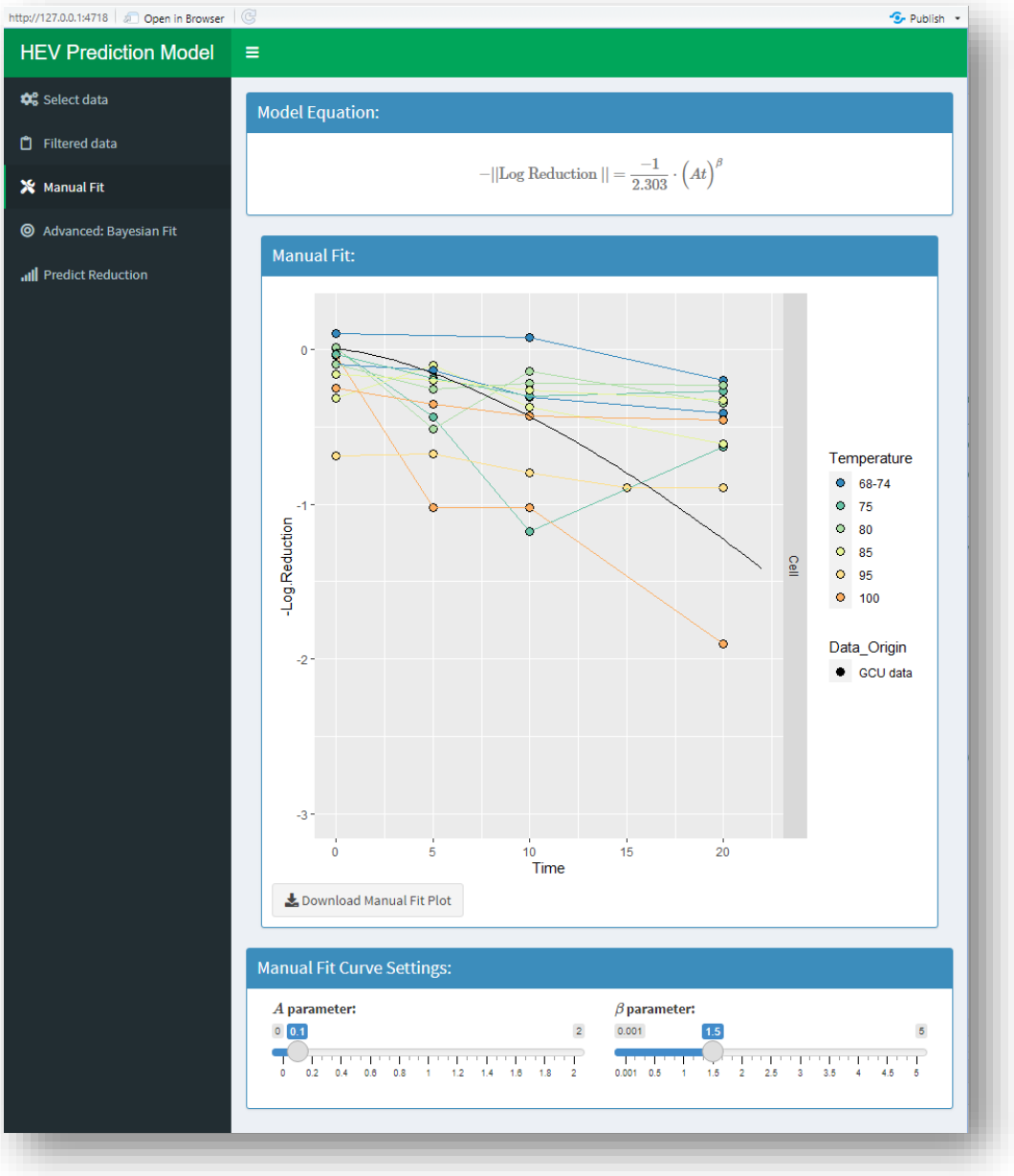

FIGURE 3: MANUAL FIT TAB

In its standard form, the model equation is a two-parameter Weibull equation, that takes the form:

$$N_t = N_0 \, exp\left[-(A\, t\,)^{\beta}\right]$$

where, N_t is the number of surviving virus particles left after a sample has been exposed to a particular temperature T (Celsius), for time t (minutes). N_0 is the number of virus particles initially, before any heating has taken place.

Adjusting the values of A and β on this page, automatically updates the corresponding model prior on the following "Advanced: Bayesian Fit" tab. ("Priors" are essentially informed guesses, as to what the true value of these parameters should be). This feature allows you to easily focus on a particular temperature, find parameter values that fit the data points for this temperature, and in turn send sensible priors to the next tab, for use within the Bayesian simulations.

The "Download" button lets you save any figures you create on this page.

10. Next the "**Advanced: Bayesian Fit**" Tab (Figure 4):

This tab fits the model equation/parameters to the data points **<u>for the selected temperature</u>**.

First, select a temperature (using the slider) to estimate A and β for, then select sensible priors (or feel free to leave these alone, if you have already estimated them using the "Manual fit" tab).
(Prior values inserted here are the mean value of the prior distributions used within simulations).

Finally: Simply hit the **"Estimate Model Parameters!"** button, and R/RJAGS will take care of the rest.

The simulations may take a while to run (Usually sub – 30seconds).

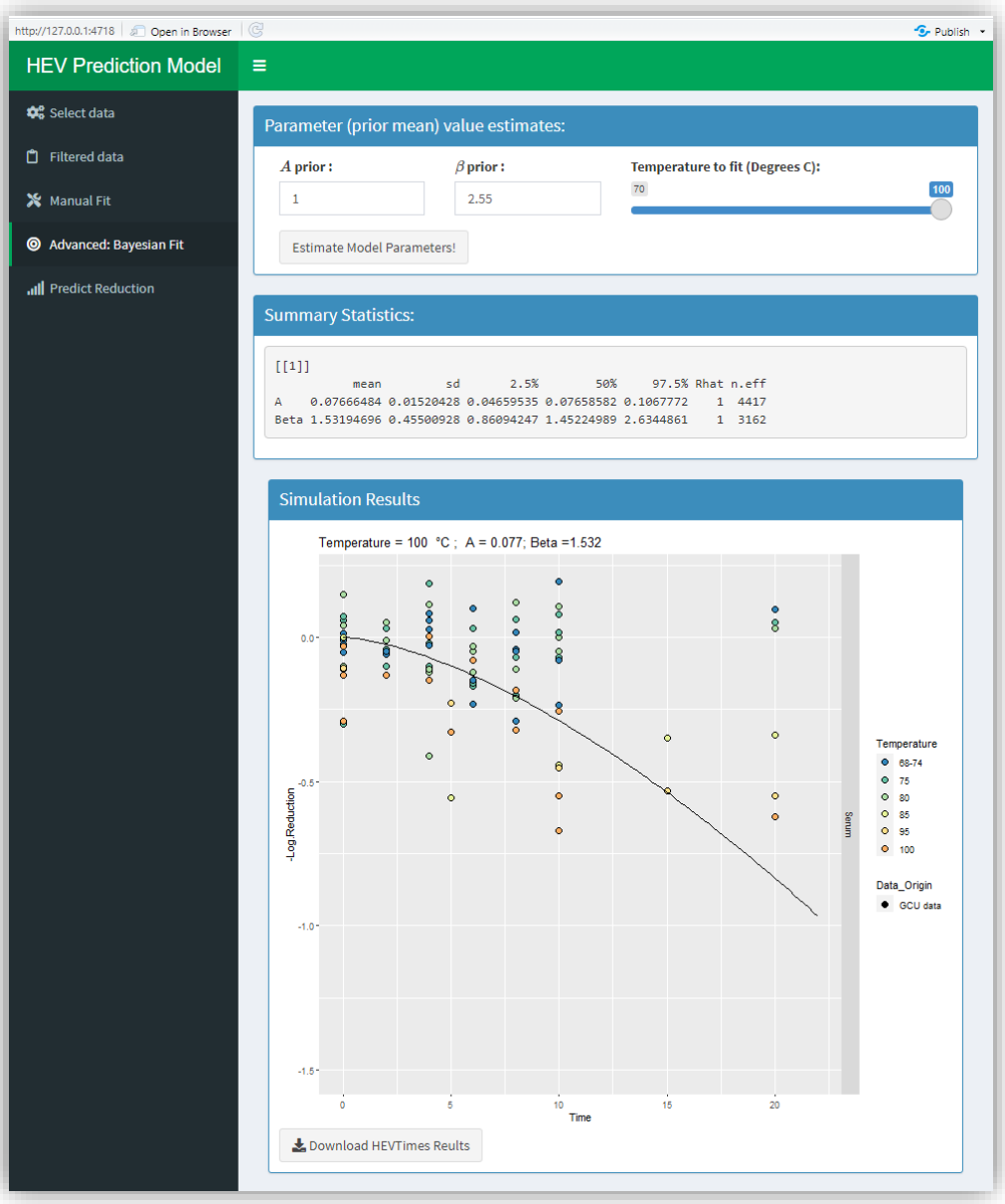

FIGURE 4: ADVANCED BAYESIAN FIT

CAUTION: Whilst each viral source is plotted individually; the Bayesian parameter estimation combines ALL selected data, from ALL selected sources, to fit A SINGLE pair of parameters (A and β), to ALL selected data. This allows data from different viral sources to be combined (if desired) to produce a single pair of parameter estimates, based on mixed viral sources.

To calculate unique parameters for each individual viral source, you must only select one single source on the initial "select data" tab, and then repeat the process for each viral source, individually.

11. **Simulation Results:**

Once the simulations have finished running, summary statistics will appear, alongside a plot of the fitted model curve.

Summary statistics include the **mean** value, **standard deviation**, **2.5% quantile**, **median** (50% quantile), and **97.5%** quantile, for the estimated model parameters A and β.

"**Rhat**" value: Convergence diagnostic statistic. If we have R-hat close to 1, it indicates convergence. Whilst if R-hat is greater than 1.05, this indicates we don't have good convergence.

"**n.eff**" value: This is the effective sample size, kept from the posterior distribution.

As a default rule, for this particular model, any n.eff value greater than 30 is good (This corresponds

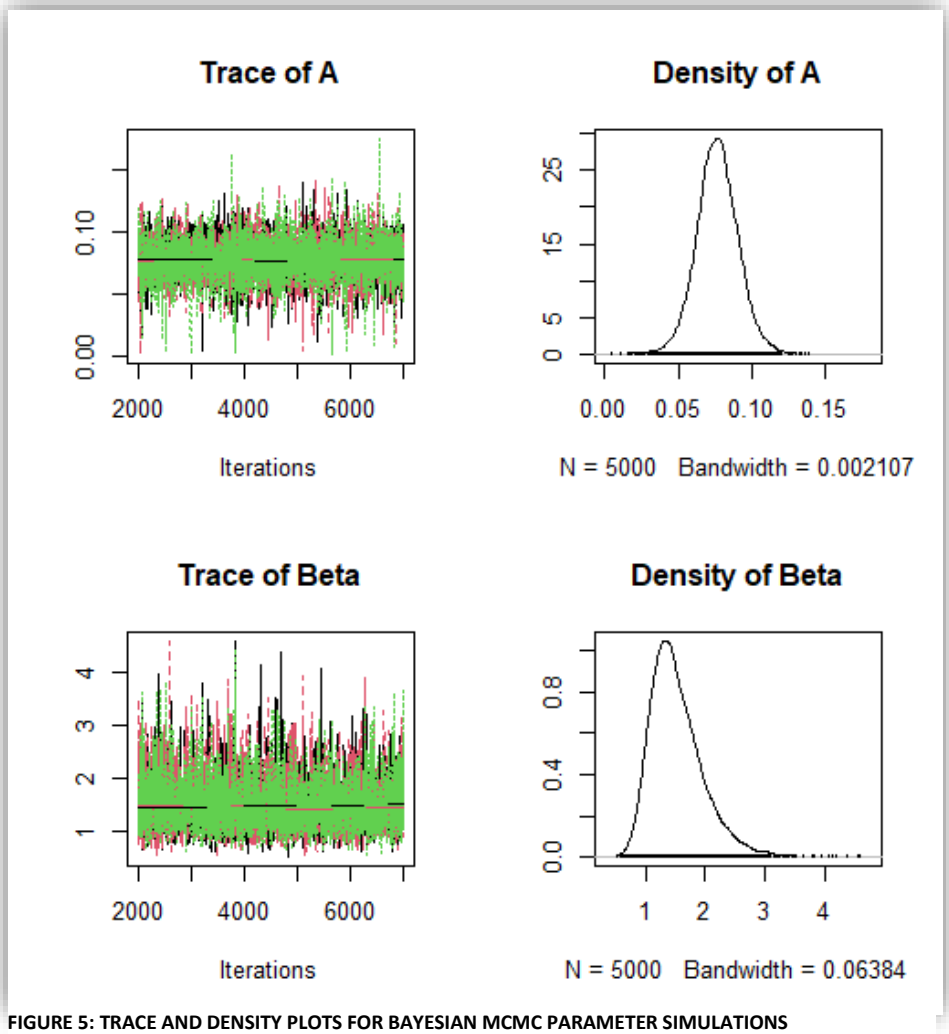

FIGURE 5: TRACE AND DENSITY PLOTS FOR BAYESIAN MCMC PARAMETER SIMULATIONS

to 10 times the number of MCMC chains used by the code). This should typically correspond to stability of the convergence (Gelman , 2013, pp. 284-290).

Note: The model uses three “Markov chains”, in its simulations. These three chains simultaneously estimate the two model parameters, at each timestep. So “convergence” refers to all three chains converging upon approximately the same value (And can be seen in the corresponding trace plots).

12. **Trace and density plots** of the MCMC simulation that has estimated the model parameters are shown within the main Rstudio window, as pictured in Figure 5/Figure 6. Depending on your current RStudio settings, you may need to click on “Plots” tab in the bottom right pane, for these to become visible.

The density plots are essentially smoothed histograms of the estimated values of each parameter (Usually termed the “posterior distribution”) and indicate the likelihood of observing the parameter value on the x-axis.

The trace plots show the convergence of the three Markov chains used to estimate the model parameters. These contain three different coloured lines, one for each chain, and we are ideally looking for these three chains to converge to the same value. This implies out estimates are good.

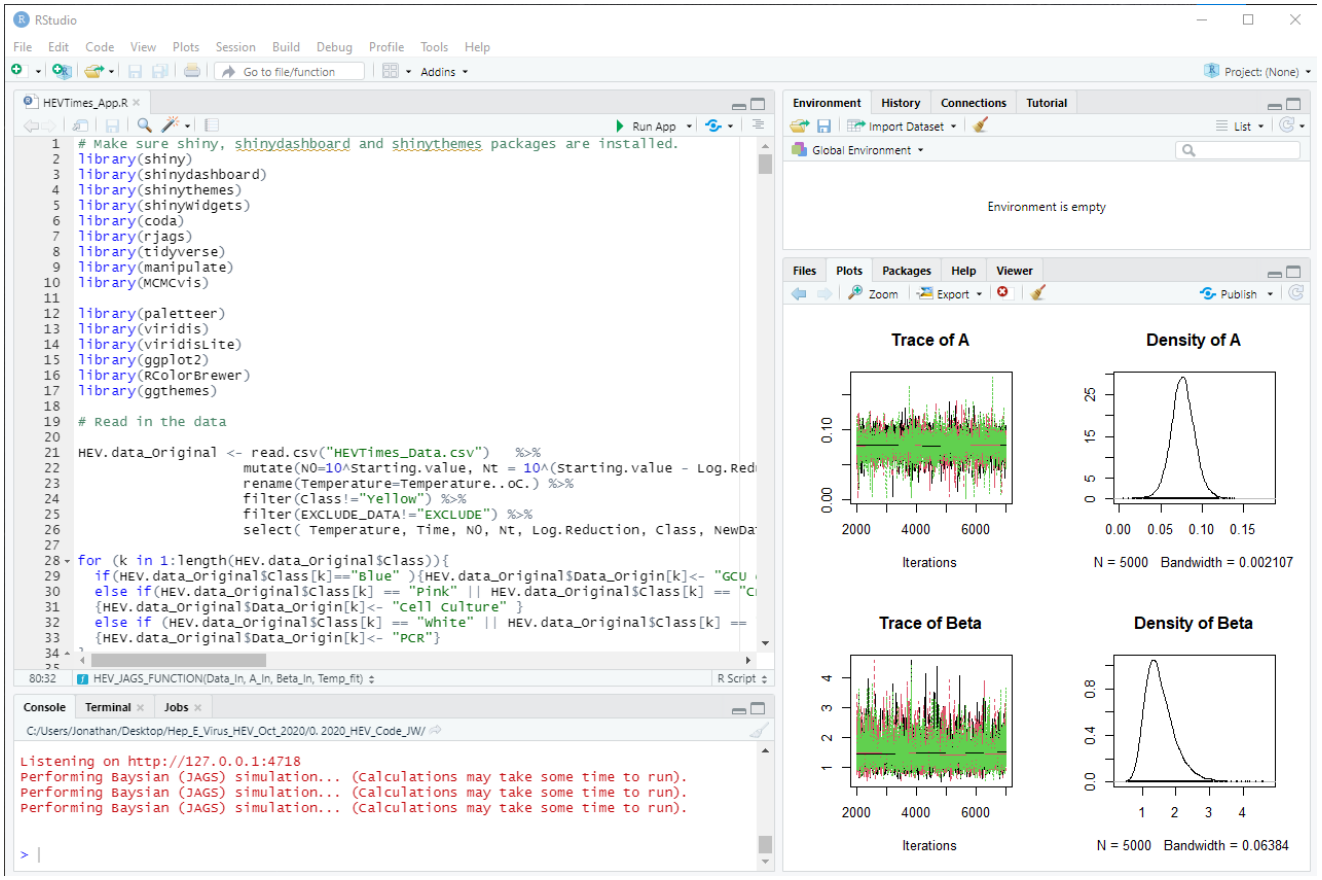

FIGURE 6: MAIN RSTUDIO WINDOW FOR REFERENCE

NOTES: MODEL USER GUIDE

1: INSTALL "R" STATISTICAL SOFTWARE

The model has been coded into a Shiny application, using the R programming language, and thus an installation of the R software (R Project, 2021) is required to run the simulations. This is an open-source programming language and software, supported by the R Foundation for Statistical Computing:

R-project.org

R software is available for all the major operating systems (Windows, MacOS, Linux, etc), so please make sure the software you install matches to your current computer's operating system. The same is true for all other software listed below.

2: INSTALL RSTUDIO SOFTWARE

R itself is a command line interface. The third party "RStudio" interface is required to tun the Shiny app. "RStudio" is available in both an open source, as well as a commercial version for organizations not able to use AGPL software (Affero General Public License). The following page should guide you through installation:

RStudio.com

3: INSTALL "JAGS" SOFTWARE

The model makes use of a Bayesian analysis technique known as Markov Chain Monte Carlo (MCMC) simulation, to remove some of the random noise found in the data whilst fitting the model equations. These are often carried out numerically using various Gibbs Sampling software variants, that have all stemmed from the original BUGS (Bayesian inference Using Gibbs Sampling) software. Here, we make use of the JAGS (Just Another Gibbs Sampler) variant of this software. This is a is a cross-platform alternative with a direct interface to R using the "rjags" R package. The model requires a working installation of the JAGS software, the most recent version is JAGS-4.3.0.exe, which can be downloaded and installed from:

mcmc-jags.sourceforge.net

4: INSTALL REQUIRED R PACKAGES

Running the attached "HEVTimes_Install_Packages.R" R-script should install all the r-packages that are required to run the app successfully. Simply open the script in RStudio, "select all", and hit "run" in the top toolbar. You only need to run this script once per computer/R installation, and the packages will always be available. You may see a message "do you want to install from sources the package which needs compilation". If this appears, you can simply choose "no", and the required packages should still instal correctly.

5: Save Data in Appropriate .csv Format

If updating the "HEVTimes_Data.csv" file with additional data. Take care to ensure the new data is saved in the exact same format as the current file, and with the exact same filename, as this filename is hard coded into the R code.

An additional, colour coded, .xlsx format Excel spreadsheet is also included, to make it easy to view the current data. It can be used as a template for the .csv file, however, requires the removal of some surplus additional columns that contain working/notes (Columns N-R). Simply open the .xlxs sheet, add new data, click "Save as…", and select ".csv". Since the surplus columns that need to be removed (Columns N-R) contain excel formulas, you will have to close the newly saved .csv file, reopen this new .csv file, and then remove additional columns. Otherwise, you may unintentionally end up with zero values in the log-reduction column (Column D).

6: Bayesian Parameter simulation notes

Model prior distributions are specified within lines 82-100 of the current version of the code; See Figure 7.

NOTE: These are written using the JAGS language, which uses slightly different syntax than the R language. So, take care if editing these!

```
HEV_JAGS_model <- " model {
 for(i in 1:length(NLR) ){
  NLR[i] ~  dnorm(Mu[i],tau )              # ASSUMES NEGATIVE LOG REDUCTION (LOG CHANGE)
                                           # IS NORMALLY DISTRIBUTED,
                                           #  with mean Mu, given by the model equation, and precision tau
                                           # Here, this "Precision" is equivalent to (1/variance).
  Mu[i] <- (-1/2.303)*((  A*Time[i]  )^ Beta )
           }

  A  ~      dunif( 0 , 2*mu1 )   #  A prior distribution = uniform distribution, with centre point given by
                                 #   the prior estimate within the app (ie. mu1 here).


  Beta ~   dunif( 0 , 2*mu2 )     # Beta prior distribution = uniform distribution, with centre point given by
                                  #   the prior estimate within the app (ie. mu2 here)

                                 # Be careful: dgamma, gamma distribution parameters are Shape and rate here.
 sigma2 ~    dgamma(1 , 4)
 tau <- 1 / ( sigma2 )           # Precision for Normal distribution used for Negative log reduction (NLR).
} "
```

FIGURE 7: RJAGS CODE EXTRACT – SPECIFYING THE MODEL FOR THE JAGS SOLVER, ALONG WITH THE CHOSEN MODEL PRIORS

MATHEMATICAL MODEL

In the above, we demonstrated the use of the Weibull mathematical model[1] (Smith R. , 1987) to successfully predict the thermal death times for Hepatitis E virus (HEV) in foodstuffs. Commonly used in survival analysis, to predict the time till death of biological organisms, and within engineering applications to predict time till failure of components parts (Smith R. L., 1991), the Weibull model has also recently been used to describe the thermal inactivation of Hepatitis A virus (HAV) in blue mussel (Bozkurt, 2014) .

The Weibull model equation takes the form

$$N_t = N_0\, exp\left[-(A\, t\,)^{\beta}\right] \qquad 1$$

where, N_t is the number of surviving virus particles left after a sample has been exposed to a particular temperature T (Celsius), for time t (minutes). N_0 is the number of particles initially, before any heating has taken place. Here, A is a constant frequency parameter and β is a constant shape parameter. These will be different for each temperature T .

Rearranging equation (1), the negative of the log reduction[2] (log change), in the number of virus particles, at time t is given by the equation:

$$\begin{aligned} -\left|\left|\text{Log Reduction}\right|\right| \;&=\; log_{10}\left[\frac{N_t}{N_0}\right] = log_{10}\left(exp\left[-(A\, t\,)^{\beta}\right]\right) \\ &= \frac{1}{2.303} ln\left(exp\left[-(A\, t\,)^{\beta}\right]\right) \\ &= \frac{-1}{2.303}(A\, t\,)^{\beta}. \end{aligned} \qquad 2$$

The model parameters A and β have then been estimated using a Bayesian approach, within the R software. This is achieved using the JAGS (Just another Gibbs Sampler) package, which uses Markov chain Monte Carlo simulation (MCMC) to estimate the parameters. Doing so allows us to account for some of the uncertainty and noise found within the literature/GCU trial data.

---

[1] The classic Weibull equation takes the form $N_t = N_0\, exp\left[-\left(\frac{t}{\alpha}\right)^{\beta}\right]$, where α and β are the scale and shape parameters, respectively (Cunha, 1998).

[2] Note: Using rules of logs: $-\left|\left|\text{Log Reduction}\right|\right| = [log_{10}[N_0] - log_{10}[N_t]] = \log_{10}[\,N_t\,] - \log_{10}[N_0] = \log_{10}\left[\frac{N_t}{N_0}\right]$.

UPDATING THE MAIN DATA FILE (HEVTIMES_DATA.CSV)

All data used by the app, is contained within the file "HEVTimes_Data.csv". This contains all literature data, plus new experimental data, and is designed to be easily edited, to allow additional data to be added to the model (or to exclude existing data).

The main HEVTimes_Data.csv file contains the following columns:

1. **Temperature (oC):**
   The temperature at which the viral sample was heated (°C).

2. **Time (HOURS):**
   The time the viral sample was heated for (Hours).

3. **Time (Mins):**
   The time the viral sample was heated for (Minutes).

   This is the main time data, that the app uses. However, as some of the literature data is recorded in hours, it is useful to have both time columns for easy conversion to minutes.

4. **Log Reduction:**
   The log-reduction observed upon heating the viral sample, for the corresponding time and temperature combination. This is calculated as:

   log-reduction = Log_{10} (Initial virus particles) – Log_{10} (Virus particles remaining at time t).

5. **Matrix:**
   Denotes the matrix the virus was obtained from.
   This is simply for reference and is not used within the app itself.
   Subsequently, any desired information can be included here, and no specific format is required.

6. **Other information:**
   Denotes any other information, about the data/virus used, that may be useful. E.g. Genotype.
   Again, this is purely for reference and is not explicitly used within the app itself, so no specific format is required.

7. **Starting value (Log!):**
   This is the log of the initial number of virus particles. I.e. Log_{10} (No. initial virus particles).

8. **Source:**
   This denotes the **viral source** that the virus was obtained from.
   All data points, with the same source, should have the same source entry in this column.
   Currently, the source options are: "**Cell**", "**Liver**", "**Soil**", "**Faecal**" and "**Serum**", and these are the names that will be displayed in the corresponding plots and tables within the app itself.
   However, the app is designed to automatically adapt, if a new source appears in this column.

Subsequently, it should be possible to add new sources, other than those already present.

9. **NewData:**
   This column specifies whether the **data origin** is new experimental data, or existing literature data. Subsequently, there are only two values this column should take:
   a. **"TRUE" –** To denote that the data is **Experimental**.
   b. **"FALSE" –** To denote that the data is from the **Literature**.

10. **EXPERIMENT_DATASET:**
    This column is **intended to group all data from the same experimental dataset together**.
    E.g. If repeated experiments were carried out, using the same experimental setup, in the same lab session, these will all be grouped together with the same "EXPERIMENTAL_DATASET" label.
    Subsequently, if the data is from the literature (not experimental), the value recorded is simply "NA".
    Similarly, if no experimental grouping is desired, this can also be set to "NA".

    The values used in this column are:
    a. **"NA"** – Literature data (or other misc. data).
    b. **"DATASET0"**
       – "Single point" experimental data.
       These are data points that are not part of a wider "time-series": In a time-series, the log reduction is measured at multiple time points in time, during the same experiment.
    c. **"DATASET1"**
       – Contains the first batch of time-series experimental results.
       This contains multiple time-series (runs), carried out, using the same experimental setup, in the same lab session.
       (Each individual time-series is then grouped by run number - see "RUN_NUMBER" below).
    d. **"DATASET2", "DATASET3" …**
       - Denotes the subsequent second and third batch of experiments.
    e. **"… DATASET100, DATASET101… "–** Future dataset labels should keep the same format.

    Note: The information in this column is mainly used to distinguish single point data (DATASET0) from timeseries data (DATASET1 and above).

11. **RUN_NUMBER:**
    This column distinguishes between different time series runs and is used to link time series points together within the plots produced by the app.

    In a time-series run, the log reduction is recorded at multiple time points in time, during the same experiment (Same virus, constant temperature). These can then be plotted together, to see the reduction in virus over time. Subsequently, ALL data from the same time series should be given the same number here, as this is the value the app uses to link them together.
    RUN_NUMER can take any whole number value in the range [1, ∞].
    If the data is not from a time-series, this can simply be left empty.

12. **EXCLUDE_DATA:**
    Easily include or exclude data from being fed into the app/model:
    a. **"KEEP" –** to use the data in the model.
    b. **"EXCLUDE" –** to exclude the data from the model.

Additional notes on updating the .csv file:

The main HEVTimes_Data.csv file can easily be edited in Microsoft Excel (or similar).

After editing the HEVTimes_Data.csv file, if you currently have the Rstudio Shiny app open, you will have to close RStudio and **restart Rstudio**, before the changes to the data will be recognised.

Similarly, **care should be taken to select the correct format of ".csv"** when saving the data as a .csv file. This is mentioned, to highlight the fact **that there are multiple variations upon the standard ".csv"** listed in the "Save as" menus within excel. You should select the option "CSV (Comma delimited) (*.csv)", as illustrated in Figure 8.

FIGURE 8: CORRECT FILE FORMAT FROM "SAVE AS" DROP DOWN MENU IN MICROSOFT EXCEL.

There is also an **additional, colour coded spreadsheet**, containing identical data, that is simply intended as an easier to read option. It is currently called "**HeVtimes_Data_update_24_2_2021.xlsx**", to reference the date the most recent data was added, and to distinguish it from the main data file. It can act as a useful template, that can subsequently be saved as a .csv file, and named "HEVTimes_Data.csv", to replace the existing one.

The filename for the main .csv data file, is hard coded into the app in line 21 of its code:

```
HEV.data_Original <- read.csv("HEVTimes_Data.csv")  %>%
```

Subsequently**, it is important to ensure that the file name (HEVTimes_Data.csv), and .csv format, both stay the same,** if adding new data. Alternatively, line 21 of the code must be edited accordingly.

REFERENCES

Bozkurt, H. D. (2014). Determination of thermal inactivation kinetics of hepatitis A virus in blue mussel (Mytilus edulis) homogenate. *Applied and Environmental Microbiology 80.10*, 3191-3197.

Cunha, L. M. (1998). Optimal experimental design for estimating the kinetic parameters of processes described by the Weibull probability distribution function.

Gelman , A. e. (2013). *Bayesian data analysis.* CRC press.

R Project. (2021). *R: A Language and Environment for Statistical Computing.* Vienna, Austria: R Core Team. Retrieved from R: A Language and Environment for Statistical: www.R-project.org

Smith, R. (1987). (1987). A Comparison of Maximum Likelihood and Bayesian Estimators for the Three-Parameter Weibull Distribution. . *Journal of the Royal Statistical Society. Series C (Applied Statistics), 36(3)*, 358-369.

Smith, R. L. (1991). Weibull regression models for reliability data. *Reliability Engineering & System Safety 34.1*, 55-76.

PART OF THE HEVTIMES PROJECT: THERMAL DEATH MODEL FOR HEPATITIS E VIRUS (HEV)