READ ME – Fit_And_Model_Age_nos

This document describes the multi-state model code. We layout the objectives of the model and then describe the purpose of each cell. Read this after **REAMME – Fit_Admission_Growth**.

Objectives: The aim of this code is to fit a multi-state model to hospital level data on COVID-19 patients. Using this fitted model, bed occupancy can be forecasted based on chosen future admission scenarios. This can aid bed management by showing how many beds are likely to be required by COVID-19 patients.

Import required packages – loads the packages required for the multi-state model

Define input variables – this is where all input parameters are specified. "file_date" should be the date indexing for input/output files, as in *Process_Data_Table*. "dtpull" should be the date of the most recent admissions in the sample (the last four days of admissions should be removed from the sample to avoid testing delays), as in *Process_Data_Table*. "forecast_start" should be the date from which we want the weekly output forecasts to start. The growth rates and initial conditions for the admissions trends in each age group are then loaded from *admissions_conditions.csv* (which is saved by the *Fit_Admissions_Growth* R script). These growth rates have an option *approx_growth*. If set to **False**, we use the age specific growth rates in the projections. If set to **True**, we apply the overall growth rate for each age group.

We then have indicator variable that need to be changed to true or false depending on whether individuals from each age group go to ICU. If 1 or fewer individuals go to ICU in an age group, set this indicator to False, as this is insufficient data to trust the fitted model.

We then need to specify how many simulations we want to run and how many MCMC samples to take. These don't need to be adjusted, but can be changed is desired. A variable "saved" can be set to "0" if we want to fit the data, or to "1" if we want to load previously fitted distributions. If the user has very few patients going to ICU, set the variable "global_dist" to be equal to 1, which will fit Weibull distributions to all transitions, instead of a combination of Burr and Weibull distributions. By default, this should be set to 0, as when sample sizes are sufficiently large this will give better results.

The final variable that can be changed is "generate_los_estimates". If set to 0, when the entire script is run it will only fit the model and generate forecasts. If set to 1, the code will fit the model and generate forecast, before simulating average patient lengths of stay and patient outcome probabilities. These are interesting to study occasionally, but might not be necessary each time forecasts are generated, and therefore this section of code can be switch off if not required.

Load length of stay data and census data – loads the length of stay data frame and hospital census generated by the *Process_Data_Table* script.

'Define states and transitions - start' – The following cell defines the states of the model and the allowed transitions. These need to be consistent with the states and transitions described in the *Process_Data_Table* script.

Return a Weibull scale parameter from the linear predictor – the model is re-parameterised to fit a linear predictor. This needs to be converted back into a scale parameter when sampling the Weibull distribution.

Define function for simulating indicative scenarios over the next few months – This function simulates the number of beds occupied over the coming months. The model simulates the occupancy independently for four different age groups: 0-25, 26-50, 51-75, 76+. For each age group, we specify the expected admissions trajectory over the coming months. We assume two potential admission trends, either constant growth or exponential growth. For the constant growth scenario, we specify the expected number of admissions per day for each age group. For the exponential growth scenario, we specify the expected number of current admissions per day and the exponential growth rate in the expected number of daily admissions. These can be determined by fitting an exponential curve through the past few weeks of admission data. The input parameter for this function is scenario (1 for exponential and 0 for constant) and nboot. nboot determines the number of simulations that we run for the admissions data, with length of stay parameters randomly sampled from the posterior distribution of the fitted model. For each patient in the admissions data, we simulate their dynamics until they either leave hospital or reach the final time point.

Simulate for posterior credibility – This function simulates a specified number of patients (npat) to determine the expected length of stay in each state and the probability of an individual in a given age group of experiencing each transition. We also specify nboot, which specifies the number of parameters we sample from the posterior. For each sample from the posterior, we run npat simulations to accurately calculate the expected dynamics.

Fitting Distributions – This cell fits the model to the observed patient transition data using MCMC. This provides a better measure of uncertainty that the bootstrapping method in the previous code, and allow us to sample different parameters from the posterior distribution to effectively propagate this uncertainty into the forecasted bed occupancy. We fit two distributions: Burr distribution for admission to ICU, admission to death and admission to discharge; and Weibull distribution for ICU to death, ICU to stepdown, and stepdown to discharge. We choose to use two distributions as this provides a better fit to the data. The cell saves the MCMC trace for each transition, so these do not need to be run every time we want to do new forecasts. This is useful, since the length of stay and proportions might not change significantly on a regular basis. However, if the user has time it is best to fit the model each week in case there has been a change in length of stay.

For transitions that are not observed in the data - For certain age groups we do not observe all transitions. Therefore, we set the related parameters to nan, so that the simulator ignores them. These will need changing to the relevant transitions in the local hospital data.

'Simulations and model output - start' – first we define nboot and npat, before calling the simulator functions.

 # Call the predictions simulator based on two scenarios – this calls the prediction simulator twice, once for each scenario

Sort probabilities of bed occupancies – based on the output of the predictions simulator, this captures the number of beds required on acute ward and ICU. This is based on three cases: 1 - 95th quantile of exponential growth scenario, 2 - 50th quantile of exponential growth scenario, and 3 - 50th quantile of constant admissions trend.

Compile bed occupancy predictions into a table – compiles the occupancy cases into a table.

'Generate figures - start' – the cells below plot figure for model output. First we plot the predicted bed occupancy. Then we plot the model bed occupancy up to the current date and compare this with the actual bed occupancy, to evaluate model performance.

 # Call the posterior simulator to give mean and uncertainty estimates for the observed lengths of stay – calls the simulator for the expected length of stays and probabilities. This function actives if the *generate_los_distribution* option is set to 1.

 # Work out probabilities of going to each state and display length of stay and probabilities plus uncertainty – work out the probability from the output of the above cell and compile the output. This output is printed as text to the screen, so needs to be copied if the user wishes to store this information.