READ ME – Fit_Admissions_Growth

The aim of this workbook is to fit an exponential growth rate to the admissions data. With a low number of admissions this can be a little tricky. It is possible to fit the whole time series of the pandemic, which would result in a time varying growth rate. The model attempts to fit as few changes to the growth rate as possible to capture the trend in the data. For analysis the current growth rate, we recommend fitting recent data, using between 28 days and 100 days of data. Often it can be worth trying both, to see which works better, as their performance can vary, particularly around turning points in the data. The length of data to use is specific in the *Process_Data_Table* script. This should provide an indication of recent daily admission trends. Taking this exponential growth rate can allow us to project future expected admissions trends.

DoubleTime – This is a function that estimates a time varying growth rate for the data using a generalised additive model. The current set up should be suitable. There are various options in the data that are redundant for fitting the admissions growth, but are present due to other uses of the code and we have not had time to tidy the code.

Main Code – in the main body of the code, we first load the required packages, followed by the data. The data should be in the following format (generated by *Process_Data_Table*):

dates; counts_1; counts_2; counts_3; counts_4; counts_5

where each counts column represents the admissions data for that age group, going 0-25, 26-50, 51-75, and 76+. The counts_5 column has the daily admissions across all age groups. In the dates column we just label the counts sequentially (e.g. from 1 to 28 for four weeks of data) rather than using the actual dates. This removes the need to convert dates to integer indicators, and the actual dates themselves are not important for this analysis.

These admissions streams are then combined into a data frame. We specify *avg_len* which is the number of days over which we will average the outputted growth rate. This is important in-case the growth rate estimate fluctuate a lot in the last few days to ensure we have a stable estimate for the forecasting model. We specify *Lag* which determines how many days to ignore. By default, this is set to 0, since we are removing two days of data from the raw input data.

We then loop the **DoubleTime** function over each column in the data frame to obtain a growth rate estimate for each age group. This assumes growth rate are independent in each group, which may hold in the short term but in reality they are correlated, and a high growth rate in one group is likely to lead to increased growth rates in other groups through community mixing.

From this loop, we output a vector *out_mean* which contains the average growth rate estimate over the last *avg_len* days for each age group. These values can then be used as input for the exponential growth rates (*rate*) in the **#### Define function for simulating indicative scenarios over the next few months** cell in the *Fit_And_Model* code. Additionally, this code outputs *initial_value.* This gives the initial number of admissions for each age group (*I0*) to be used in the **#### Define function for simulating indicative scenarios over the next few months** cell in the *Fit_And_Model* code. These are saved *as admissions_conditions.csv*, which is loaded into the python script.

This code also generates figures showing the growth rate estimates over time and the fitted model compared to the data for each age group. This is to allow model performance to be evaluated. This is important, as due to the small sample sizes and small case counts, the model performance may not be reliable, so we need to sense check the outputs.

Sense check

This method can be quite temperamental, especially around the end of the data window. To sense check the output, compare the age-based growth rates to the total admissions growth. These should be reasonable consistent, in that the overall growth rate should look like a weighted average of the age-based growth rates. When the output of *Fit_And_Model* has been produced, a further sense check can be checking the *modelfit_(filedate)_burr_growth* figure. In this figure, the forecasted trend (black curves) should be a continuation of the trend in the actual data (red curve).