Did COVID-19 infections decline before UK lockdown?

Simon N. Wood
University of Edinburgh, UK. simon.wood@ed.ac.uk

September 21, 2020

Abstract

The number of new infections per day is a key quantity for effective epidemic management. It can be estimated by testing of random population samples. Without such direct epidemiological measurement, other approaches are required to infer whether the number of new cases is likely to be increasing or decreasing: for example, estimating the pathogen reproductive rate, R, using data gathered from the clinical response to the disease. For COVID-19 (SARS-CoV-2) such R estimation is heavily dependent on modelling assumptions, because the available clinical case data are opportunistic observational data subject to severe temporal confounding. Given this difficulty it is useful to reconstruct the time course of infections from the least compromised available data, using minimal prior assumptions. A Bayesian inverse problem approach applied to UK data on COVID-19 deaths and the disease duration distribution suggests that infections were in decline before full UK lockdown (24 March 2020), and that infections in Sweden started to decline only a day or two later. An analysis of UK data using the model of Flaxman et al. (2020) gives the same result under relaxation of its prior assumptions on R.

Introduction

Clinical data on the number of cases of COVID-19 (SARS-CoV-2) are subject to severe temporal confounding, as the rate of testing and criteria for testing have been changing rapidly on the same time scale as the infections, particularly in the early weeks and months of the epidemic. Because the ascertainment fraction is changing and unknown, the data can clearly not be used to infer the actual number of infections. Neither, under normal circumstances, would statisticians recommend attempting to estimate the reproductive rate of the pathogen from such data, since given the data problems the estimates must necessarily be driven strongly by the modelling assumptions. Indeed generically it is often very difficult to infer epidemiological parameters from clinical data, without the results being informed as much by the prior beliefs encoded in the model as by the data (e.g. Wood et al., 2020).

The exception is when clinical data directly measure the quantity of epidemiological interest. This is the case for deaths with COVID-19 and for fatal disease duration. While not perfect, these data are far less compromised than the data on 'cases'. Deaths are reliably recorded, clinical grounds for suspecting COVID-19 are clear, and good records are kept for fatal cases. It is of some interest to establish what these high quality data imply about the time course of infections, without strong modelling assumptions.

Two types of daily death data are available. Daily reported deaths (e.g. Worldometer, 2020) typically show marked weekly fluctuations as a result of weekly patterns in reporting delays, and may exclude deaths in some locations (such as nursing homes). Registered death data, such as the ONS data in the UK (Office for National Statistics, 2020), contain deaths in all locations and record exact date of death. NHS England[1] publishes equivalent data for hospital deaths in England. The weekly cycle is less pronounced

[1] www.england.nhs.uk/statistics/statistical-work-areas/covid-19-daily-deaths/

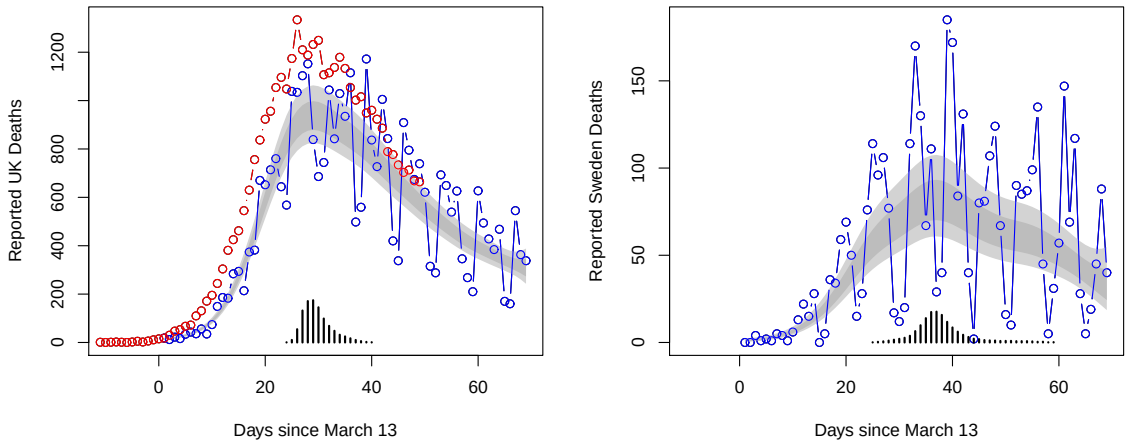

Figure 1: Daily reported deaths with COVID-19 (blue) in the UK (left) and Sweden (right) since March 13th, based on immediate reporting. In red is the UK ONS data for England and Wales for all locations of death by registered day of death, illustrating the lag in reported deaths. The grey regions illustrate 68 and 95% confidence regions for the underlying reported death rate from model (1). The bar charts are proportional to the posterior distribution of day of peak underlying rate according to model (1). Full UK lock down started on day 11 (March 24). Sweden implemented targeted measures short of lock down.

in these data, but their release is necessarily delayed relative to the daily reported deaths. Figure 1 shows daily reported data for the UK and Sweden, from a time point when reporting effects were quite marked.

Data on the incubation period from infection to onset of symptoms are analysed in many papers, for example Lauer et al. (2020) found that the period is 2 to 11 days for 95% of people, with a median of 5.2 days. A meta-analysis by McAloon et al. (2020) suggests a log-normal distribution with log scale mean and standard deviation of 1.63 and 0.50.

Several studies estimate the distribution of time from onset of symptoms to death, while properly controlling for the right truncation in the fatal duration data. Verity et al. (2020) found that the distribution of time from onset of symptoms to death for fatal cases can be modelled by a gamma density with mean 17.8 and standard deviation 8.44, based on 24 patients from Wuhan. Wu et al. (2020) suggested a gamma density model with mean 20 and standard deviation 10 based on 41 patients from Wuhan. Linton et al. (2020) found that a lognormal model offers a slightly better fit, and estimated a mean of 20.2 days and standard deviation of 11.6 days from 34 patients internationally.

Data for England are available in the CHESS[2] database, access to which is restricted to particular research groups under strict conditions. With the kind help of Robert Verity from Imperial College I was able to access information on the distribution of fatal disease durations for 3274 deaths that occurred before 10 June 2020 with recorded symptom onset before 1 May. The information provided was a bar chart of the duration distribution by day, on condition that only the information about the model fitted to the data be distributed further. The data were not filtered to remove hospital acquired infections, but it was not possible to obtain data only for those with onset before hospitalization. This is problematic for two reasons. Firstly, for inferring the time course of community acquired fatal infections it is the distribution of fatal disease durations for community acquired infections that is required, which the raw data do not provide: for example, they contain substantial proportions of durations of 1-3 days that appear clinically implausible for deaths from community acquired COVID-19 (see, e.g. Huang et al., 2020; Wang et al., 2020; Zhou et al., 2020; Tay et al., 2020). Secondly the raw data are from a relatively

[2]COVID-19 Hospitalisations in England Surveillance System

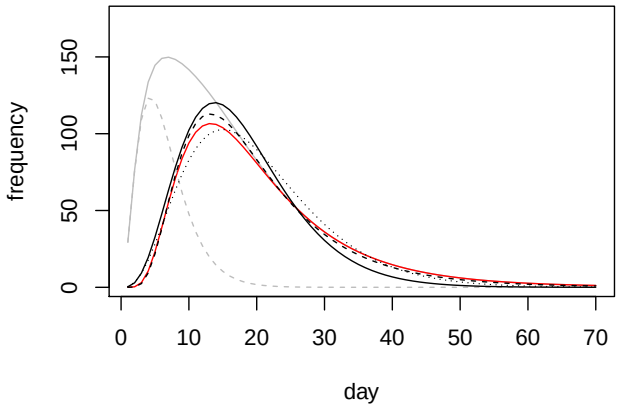

Figure 2: Onset to death duration distribution models. The red curve is the log-normal mixture component for community acquired infection fitted to the CHESS data, the dashed grey curve is the gamma mixture component representing hospital acquired infection and the continuous grey curve the combined model. The combined model is not directly usable: see text. The black curves are: continuous Verity et al. (2020); dashed Linton et al. (2020); dotted Wu et al. (2020). The mixture model was estimated by maximum likelihood, with the hospital acquired mixture proportion reduced until the profiled log likelihood was reduced to 4 below the MLE, to obtain the shortest mean community acquired duration consistent with the data under this model. The black curves in no way inform the red curve in the fitting.

small proportion of the total deaths. It is very unlikely that the ratio of hospital to community acquired infections in this sample is representative: for hospital acquired infections the onset of symptoms is presumably almost always known, and hence more likely to be recorded than for community acquired infections. This makes the raw distribution unrepresentative of the distribution for all deaths and also not usefully informative about the proportion of all deaths that are from hospital acquired infection. Note also that without more extensive data access it is not possible to rule out that some proportion of what appear to be hospital acquired infections really represent data problems (for example recording onset day as hospital admission day).

To deal with these issues a two component mixture model was fitted to data digitized from the bar chart, consisting of a gamma distribution (representing hospital acquired infections) and a log-normal distribution (representing community acquired infections). Parameterization was such that the log-normal had the longer mean duration. The higher the gamma mixture proportion the larger the log-normal mean. To find the shortest mean community acquired duration defensible from the data, the gamma mixture proportion was reduced to the point at which the log likelihood was about 4 below the MLE (decreasing further decreases the log-likelihood sharply, pushes a χ^2 goodness of fit statistic into the significant range, and starts to suggest rather high probabilities of very short disease durations for the log-normal mixture component). This point has about 0.7 of the mixture contributed by the community infection component. The resulting log-normal community infection fit has a mean of 21 days and a standard deviation of 12.7. Longer durations would be slightly more consistent with the data under the mixture model, but given the aims of this paper it is better to use conservatively short estimates here. Figure 2 shows the various estimated distributions over the duration range observed in the CHESS data. The log-normal model has an earlier mode, but longer tail, than the Verity et al. (2020) model used in earlier versions of this paper.

Assuming independence of incubation period and onset to death period, the preceding fit and the McAloon et al. (2020) incubation period imply that the infection-to-death distribution can be well mod-

elled by a log-normal distribution with log scale mean and standard deviation of 3.19 and 0.44, respectively. That is a mean of 26.8 days and standard deviation of 12.4 days.

Models

Let y_i denote the deaths or reported deaths on day i. Assume that y_i follows a negative binomial distribution with mean μ_i and variance $\mu_i + \mu_i^2/\theta$. Then let

$$\log(\mu_i) = f(i) + f_w(d_i) \tag{1}$$

where f is a smooth function of time measured in days, and f_w is a zero mean cyclic smooth function of day of the week, $d_i \in \{1, 2, \ldots, 7\}$, set up so that $f_w^{[k]}(0) = f_w^{[k]}(7)$, where $k = 0, 1$ or 2 denotes order of derivative. $f(i)$ represents the underlying log death rate, while f_w describes the weekly variation about that rate. The functions f and f_w can be represented using splines with associated smoothing penalties $\lambda \int f''(t)^2 dt$ and $\lambda_w \int f_w''(d)^2 dd$. Hyper-parameters λ and λ_w control the smoothness of the functions, and can be estimated as part of model fitting using a standard empirical Bayes approach (see methods). This model provides a good fit to both the reported deaths and ONS data. As expected f_w is greatly attenuated for the ONS data (it vanishes for Swedish exact death date data).

To estimate the daily infection profile the model must be extended. Consider expressing $f(i)$ in terms of the time course of earlier infections. Let $f_c(i)$ be the function describing the variation in the number of eventually fatal infections over time. Let $\mathbf{B}$ be the square matrix such that $B_{ij} = \gamma(i - j + 1)$ if $i \geq j$ and 0 otherwise, where γ denotes the infection-to-death log normal density given above. If $\mathbf{f}_c = [f_c(0), f_c(1), \ldots]^\mathsf{T}$ and $\boldsymbol{\delta} = [\delta(1), \delta(2), \ldots]^\mathsf{T}$ then $\boldsymbol{\delta} = \mathbf{B}\mathbf{f}_c$, where $\delta(i)$ is the expected number of deaths on day i. $\log f_c(i)$ can be represented using a spline basis, again with a cubic spline penalty. The final model is then obtained by simply substituting $f(i) = \log \delta(i)$ into (1). $\mathbf{B}$ is rank deficient, so inferring f_c can be viewed as an inverse problem: without regularization multiple solutions that oscillate from day-to-day are possible. This ambiguity is removed by the smoothing penalty on $\log f_c$.

As described in the methods appendix, inference about f_c was conducted using an empirical Bayes approach followed by an efficient Markov Chain Monte Carlo step to refine the results (and was also checked using a fully Bayesian Gibbs sampling approach). This exploits the fact that smoothing penalties can be induced by the adoption of appropriate Gaussian smoothing priors. It is necessary to infer f_c over a considerable period before the first death occurs. 40 days is clearly sufficient given the form of γ. In fact it makes sense to reduce this interval, after inspecting a pilot run, to avoid a lengthy initial period of zero fatal cases, consequent lack of identifiability of $\log f_c$ and poor MCMC mixing. On this basis a 20 day initial period is more than sufficient. For stable inference it also makes sense to explicitly include in the death data the fact that no deaths were observed in this initial period.

Results

Figure 3 shows the results of applying the model to the Office for National Statistics daily COVID-19 death data for England and Wales, to the NHS England hospital data and to the Folkhälsomyndigheten daily death data for Sweden[3]. I failed to find exact death date data for Scotland. The ONS and NHS data were truncated to remove the latest data highlighted as still provisional. The most notable feature of the results is that fatal infections are inferred to be in substantial decline before full lockdown. Sweden appears most likely to have peaked only one or two days later. The results also emphasise the fact that the infection trajectory is not simply a time shifted version of the death trajectory (assuming it was might lead to unwarranted delay in easing lockdown, for example). The difference in timing and shape of the

[3] `experience.arcgis.com/experience/09f821667ce64bf7be6f9f87457ed9aa`.

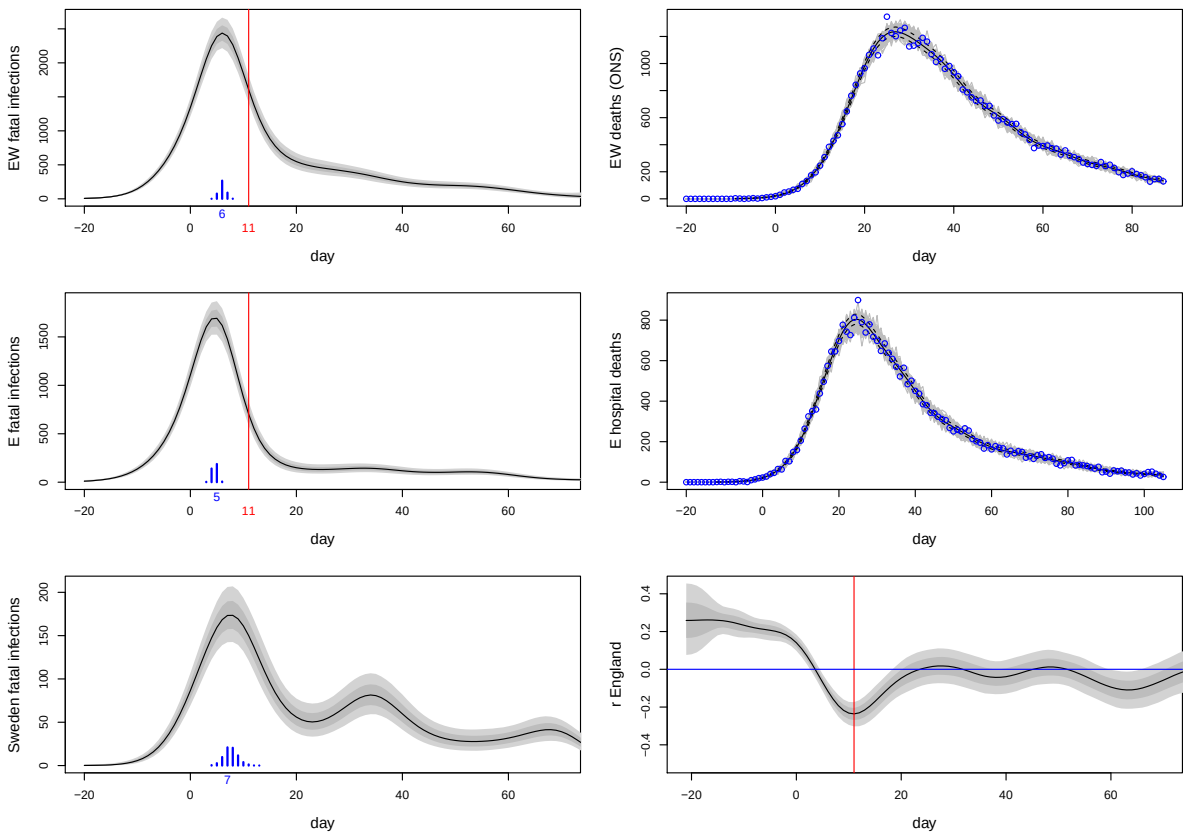

Figure 3: In all plots black curves show the posterior median while light grey and dark grey regions show 95% and 68% confidence regions, respectively. Day 0 is 13th March 2020, and the vertical red line marks the first day of UK lockdown. Top left: Inferred daily fatal infection rate, f_c, for England and Wales. The scaled blue barchart shows the posterior distribution for day of peak infection with the peak day labelled. Top right: Consistency check. In grey are 100 sets of death data simulated forward from the inferred median fatal infection profile. Blue symbols are the ONS daily death data for England and Wales on which inference is based. The dashed curves are 95% confidence intervals for underlying death rate estimated by direct fitting of (1). Middle row: As top row, but using the NHS England daily hospital death data. Note that the inferred infection trajectories are substantially different to time lagged versions of the deaths trajectories. Bottom left: The inferred fatal infection profile for Sweden, based on exact death date data, plotted in the same way as for England and Wales. Bottom right: Inferred instantaneous intrinsic growth rate of infections, r, for England based on the middle row fits. The blue line is at $r = 0$, the boundary between increase and decline in the daily infections.

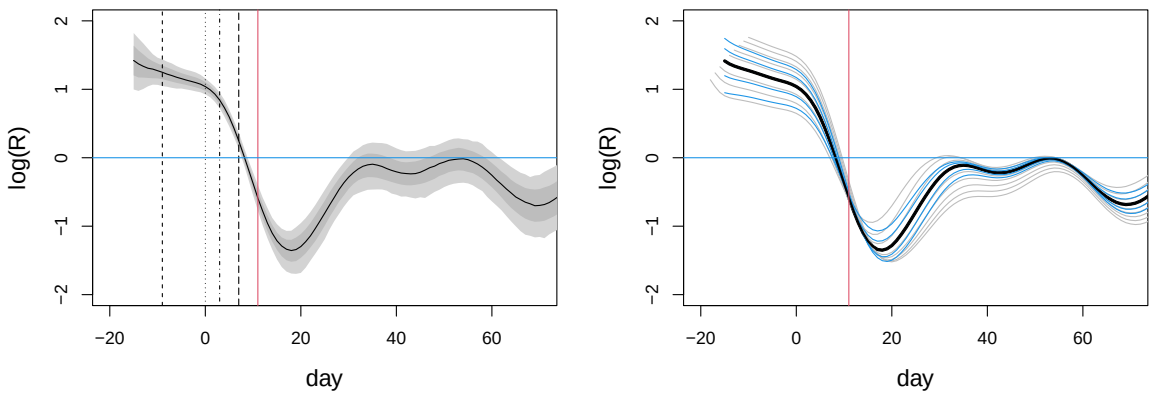

Figure 4: Left: Estimates and confidence bands for the pathogen reproductive rate, R, from a simple SEIR model given the inferred infection profile (incidence), f_c. The assumed mean time to infectivity was $1/\gamma = 3$ days and the mean infectivity duration was $1/\delta = 5$ days. The vertical bars show policy change dates in March 2020: start of public information campaign (4th, dashed); start of voluntary self isolation if symptomatic (13th, dotted); start of government promotion of voluntary social distancing, home working where possible and longer self isolation (16th, dash-dot); leisure industry and school closures (20th, long dash); full lockdown (24th, red). Given the rapidity of policy change relative to the epidemic's dynamic time scale, and government policy sometimes lagging behaviour, casual over interpretation of these timings should be avoided. Right: sensitivity analysis. Blue – time to infectivity was varied from 1 to 5 days. Grey – duration of infectivity was varied from 2 to 10 days. Logs are natural. R was well below 1 before full lockdown, but fell further after it.

inferred profile between the ONS and NHS data reflects the fact that the latter contain care home data. There is an argument for preferring hospital data for inferring community fatal infections, in that the care home epidemic is now known to have special features with at least some of the infection not coming from normal community transmission. In addition care home deaths are often attributed to COVID-19 without a test, especially since death certification guidelines were changed to encourage reporting of suspected, rather than confirmed COVID-19 deaths. The care home data therefore have some under-reporting of Covid deaths, followed by over-reporting (the signal of this is visible in ONS data in the change in non-Covid pneumonia deaths being reported, relative to normal, for example).

Taken together the results for England and Wales and Sweden raise the questions of firstly whether full lockdown was necessary to avoid health service overload, or whether more limited measures might have been effective (calling into question the implicit decision to heavily discount future life loss consequential on full lockdown in decision making – see Discussion), and secondly whether the several month duration of full lockdown was appropriate. These emphasise the desirability of statistically well founded direct measurement of epidemic size through randomized testing. Had such testing being carried out leading up to lockdown it would have been clearer if the measures preceding lockdown[4] were working, or whether stronger restrictions were needed. Similarly such testing might have given earlier indication of when lockdown could be eased. Instead management was reliant on a complex modelling synthesis of expert judgement and problematic clinical case data. Less statistically problematic reconstructions, like the one presented here, are clearly only possible weeks after the fact. Note that while it is natural to interpret these fatal infection trajectories as proportional to the overall infection trajectories, that will only be the case if the infection fatality rate is constant over time. While this seems to be the usual assumption, it comes with no guarantees.

[4]Fig 4 and `www.health.org.uk/news-and-comment/charts-and-infographics/covid-19-policy-tracker`.

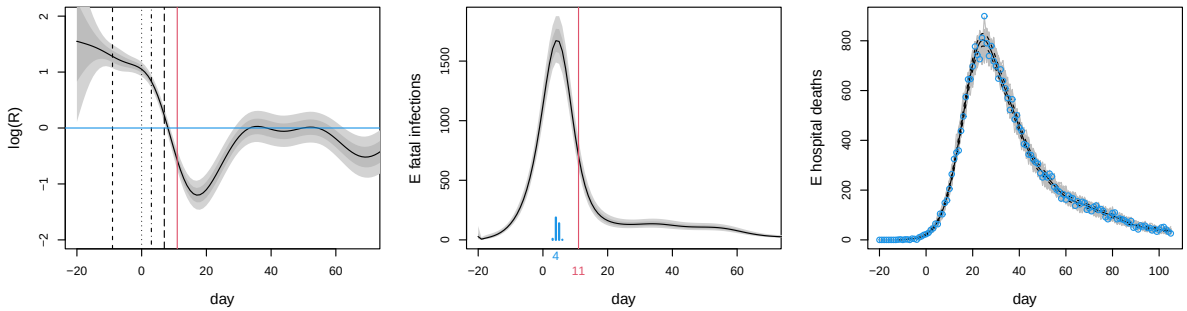

Figure 5: Results from the epidemic model of Flaxman et al. (2020), with the assumptions on R relaxed: $\log R$ is assumed smooth and continuous. Left: the inferred R from fitting the NHS hospital data. The inferred R trajectory is similar to the one shown in figure 4, despite the different model structure. Middle: the corresponding fatal infection profile. Right: the simple sanity check as in figure 3.

Figure 3 also plots the instantaneous intrinsic growth rate of daily infections, r, (the time derivative of $\log f_c$). Daily infections increase for $r > 0$. Over-interpretation of this quantity should be avoided: conceptually it relates to a single well mixed population, but the population was in fact stratified in several ways at lockdown.

Much public debate has focused on the pathogen reproductive rate, R, and in theory it is possible for a decline in the rate of infections to be only temporary as a result of R dropping but remaining above one. Could it be that the declines in f_c seen before lockdown were of this short term type, and that renewed increase would therefore have occurred without full lockdown? The answer appears to be no. R is exceptionally difficult to measure directly, but given an epidemic model it can be directly inferred from the reconstructed infection profile. For example consider a simple SEIR model: $\dot{S} = -\beta SI$, $\dot{E} = \beta SI - \gamma E$, $\dot{I} = \gamma E - \delta I$ (here δI is the rate of recovery *or* progression to serious disease). $\hat{f}_c$ is a direct estimate of βSI (to within a constant of proportionality), so by solving

$$\dot{E} = \hat{f}_c - \gamma E, \qquad \dot{I} = \gamma E - \delta I$$

(from 0 initial conditions) the direct estimate $R = f_c/(I\delta)$ is readily computed (any constant of proportionality cancels in R). Figure 4 shows the results using $\hat{f}_c$ for the English hospital data for plausible values of average time to infectivity of $1/\gamma = 3$ days and mean duration of infectiousness of $1/\delta = 5$ days, along with sensitivity analysis for these values. R appears to be below 1 before full lockdown.

Since this work was first carried out in late April 2020 other papers have been published based on analysis of death trajectories. Notably the results in Flaxman et al. (2020) apparently contradict figure 4: the authors concluded that only after full lockdown did R drop below 1, and that fatal infections continued to increase up until the eve of full lockdown. Flaxman et al. (2020) used the same Verity et al. (2020) fatal disease duration distribution employed in earlier versions of this paper, so the difference in results does not lie there. To describe the epidemic dynamics Flaxman *et al.* use a simple single compartment discrete renewal model. Within that model they assume that R is constant between the imposition of interventions, but can undergo a step change at each intervention: the steps are free model parameters. This model for R is quite restrictive. In particular it does not allow R to change after lockdown, despite the fact that at lockdown the population has been stratified in a way that the renewal model does not represent, so that some compensating flexibility in R is likely to be required to avoid modelling artefacts. At the same time the model is rather underdetermined preceding lockdown, because of the frequent intervention changes. This indeterminacy in the model is addressed by using a sparsity promoting prior on the step changes in R, which favours few larger changes, rather than several smaller changes (see the supplementary material for Flaxman *et al.* for a description of this prior). When using

the model to simultaneously model multiple European countries there is a further assumption that the intervention effects are the same for all countries (despite the different order of their implementation) and that only the lockdown effect varies between countries. It seems likely to be difficult to pick up effects of the interventions preceding lockdown from such a model structure.

It is straightforward to use the methods here with the Flaxman et al. (2020) renewal model, and to relax their assumptions about R, by again representing $\log R$ as a smoothing spline and using the renewal model to map the spline coefficients to f_c (see the Methods section for further details). Hence an assumption that $\log R$ is some smooth continuous function is substituted in place of the assumption that R is a step function with possible steps at intervention times and a prior favouring few steps. Allowing R to change continuously should mitigate the fact that the epidemic model does not attempt to represent the large change in the structure of the susceptible population (into locked down and health/key worker sub-populations) that must accompany lockdown. As already mentioned the fatal disease duration model is essentially the same between this paper and Flaxman et al. (2020). The results from the relaxed Flaxman *et al.* model applied to NHS hospital data are shown in figure 5. The relaxation of the assumptions on R brings the results into alignment with those in the rest of this paper, at least for the UK.

Note that r and R estimates based on reported case data are unlikely to match the estimates in figures 3 to 5. Especially early on, testing for cases in the UK has been focused on workers and patients in the health system, not the general population: without strong modelling assumptions the estimates are likely to reflect the epidemic in the sampled population (although even there the sampling is opportunistic) rather than the general population. There is also little reason to expect the estimates to correspond with simulation model based predictions of R that have not been statistically validated.

Model checking

While standard residual checks indicate no problem with the model from the point of view of statistical fit, there are two issues which could potentially undermine the conclusions.

The first relates to the infection to death interval distribution and the fact that the death data contain an unknown proportion of patients whose infection was hospital acquired. These patients are likely to have had shorter disease durations, since they were already sufficiently unwell or frail to be in hospital, and the mixture model shown in figure 2 is consistent with this. This paper has inferred when the fatal infections would have occurred if they were all community generated, since it is the community infections that are of interest with respect to the effects of lockdown, social distancing etc. Without knowing even the proportion of deaths from hospital acquired infection it is anyway not possible to do otherwise.

The presence of hospital infections in the death data will bias inference about the dynamics of community fatal infections if it substantially changes the shape of the deaths profile, relative to what would have occurred without hospital infection. Broadly, if the profile of hospital acquired infection deaths peaked earlier than the overall profile, then the community infection peak will be estimated to be earlier than it should be (since the true community infection death peak is then later). Conversely, if the hospital acquired infection deaths peaked later, then the community infection peak will estimated as being later than it should be. The degree of bias will depend on the proportion of hospital acquired infections and the degree of mismatch in timings. It is difficult to judge which alternative is more likely: standard epidemiological modelling assumptions would imply that the more community acquired cases are hospitalised the more hospital infections would occur and that hospital infections will lag community cases. But against this, the hospital acquired disease durations appear to be substantially shorter. In any case the proportion of hospital acquired infections in the death series would have to be quite high for the issue to substantially modify the conclusions.

It could also be that the assumed community acquired fatal disease duration distribution is systematically wrong. Given the close correspondence between the inferred community infection mixture compo-

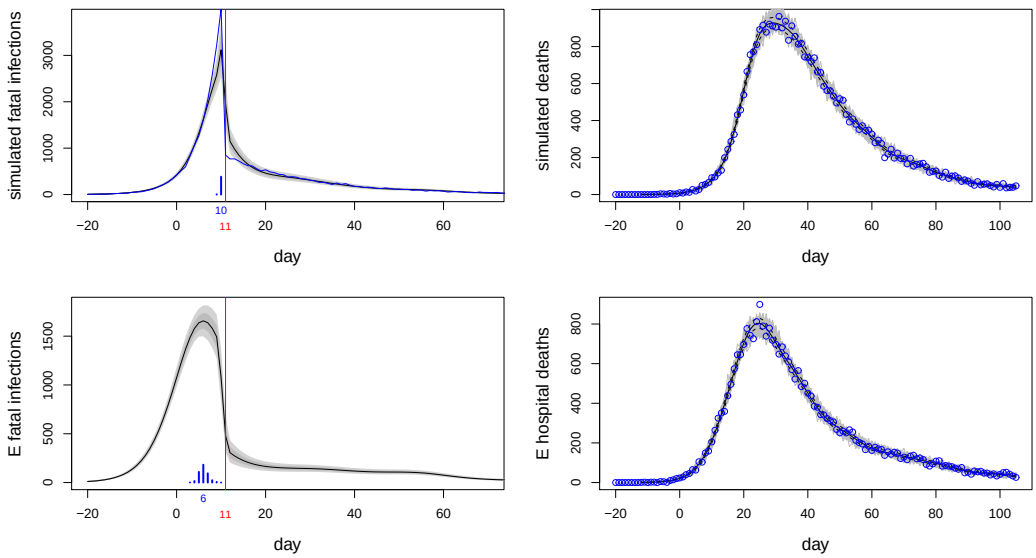

Figure 6: Model checking plots in which the smoothness assumptions are relaxed around lockdown by a time dilation, in order to allow accurate capture of any extremely discontinuous infection profile in this region. The top row shows the method reconstructing an extreme simulation scenario in which there was no reduction in transmission rate up until lockdown, and then an instantaneous drop. Left: the reconstruction (plot meaning as figure 3) with the true simulated daily infections shown in blue. Right: forward simulation from the median profile as in figure 3. The blue symbols are the simulated death data used for inference. The bottom row is for the NHS England hospital data under the time dilated model. Even this model deliberately modified to promote a very abrupt change at lockdown suggests that the infection rate was probably declining before lockdown.

nent and previous studies (see figure 2) this does not seem to be the most likely source of error, but it is difficult to be certain without access to the data on community acquired infection durations only. Sensitivity analysis modifying the mean fatal infection duration by a few days tends to shift the fatal infection profile by the same amount, with shorter duration giving a later peak, but recall that the mean duration used already corresponds to a quite short duration, given the data. Age dependency in the duration distribution coupled with shifts in the age structure of deaths over time could also be problematic.

The second issue is whether the smoothing penalty on $\log f_c$ would lead to systematic mis-timing of the estimated peak under the scenario of a very asymmetric peak in the true infection profile around lockdown. To investigate this, data were simulated from a model in which the underlying infection rate increased geometrically, doubling every 3 days until lockdown, when the rate dropped immediately to 0.2 of its peak value, shrinking thereafter by 5% per day. Fatal infections were simulated as Poisson deviates with the given underlying rate. This model is an extreme scenario, in which measures prior to full lockdown had no effect, and the effect of lockdown was instant, as if the locked down population (i.e. those not in essential work) had isolated alone, rather than increasing their contact with members of their household while drastically reducing it with everyone else. However it is the scenario implicit in much public discussion in the UK, at least at the time that this work was originally conducted. Under this scenario, the method does indeed tend to incorrectly estimate the infection peak as 2 to 3 days before lockdown, rather than the day before, as it struggles to accommodate the drop.

The naive approach to this issue is to introduce a parameter at lockdown representing an instantaneous drop in infections. However doing so introduces a very strong structural assumption into the model, undermining the aim of avoiding strong assumptions. This approach also has the serious side effect of introducing non-parametric smoothing boundary effects on both sides of the break. These boundary effects severely compromise inference in the most interesting region of the infection profile, while simultaneously increasing the importance of the structural assumption at the expense of the data.

Indeed when such a model is built it estimates a large drop even from data simulated from a smooth infection profile. It also estimates such a drop if the drop's location is moved (for simulated or real data).

A better approach is to use a smooth time-dilation to relax, but not eliminate, the model smoothness assumptions in the vicinity of lockdown. The dilation is made sufficient that the model can accurately capture the extreme scenario in the simulation, but without imposing a break and boundary effects. In particular f_c and its smoothing penalty are computed with respect to a version of time which makes the day before, of and after lockdown count as 3.5, 6 and 3.5 days, respectively. Obviously regular un-dilated time is used for mapping infections to deaths. For the extreme simulation, the model then correctly gives most posterior probability to the day before lockdown as the peak. In contrast the same model for the real data has very low probability of the peak being the day before lockdown rather than earlier.

Figure 6 shows the results from fitting the time dilated model to the extreme simulation scenario and to the NHS England hospital data. Even this model, deliberately modified to favour a very abrupt change at lockdown, suggests that infections started to decline before lockdown, with the most likely day for the peak only 1 day later than with the un-dilated model.

Discussion

This paper does not prove that the peak in fatal infections in England and Wales preceded lockdown by several days. Indeed the failure to undertake the sampling that could have gathered data to directly measure infections early in the epidemic means that it will never be possible to be certain about timings, given the substantial biases in clinical data other than deaths and fatal disease duration. What the results show is that, in the absence of strong assumptions, the currently most reliable data strongly suggest that the decline in infections in England and Wales began before full lockdown, and that community infections, unlike deaths, were probably at a low level well before lockdown was eased. Furthermore, such a scenario would be consistent with the infection profile in Sweden, which began its decline in fatal infections shortly after the UK, but did so on the basis of measures well short of full lockdown.

These facts may have implications for the policies to be adopted in subsequent infection waves, particularly given the peculiar ethical issues associated with lockdown. For example, a plausible estimate of the life loss burden from an unmitigated COVID-19 epidemic in the UK is about 2 weeks per person[5]. A plausible *lower bound* on the UK life loss from the 2008 financial crisis and its aftermath is 7 weeks per person[6]. The economic shock from lockdown is substantially larger than 2008: Bank of England projections suggest the largest shock for 100 or 300 years. Viewed another way, stringent suppression measures might save 2 million UK life years, but the same UK population was on course to suffer around 200 million lost life years associated with economic deprivation and inequality before the COVID-19 crisis[7]: carefully balanced policy is required to ensure that the suppression measures do not exacerbate this by much more than one percent and lead to a net loss of life. Similarly the implied willingness to pay to save a life year from COVID-19 appears to be an order of magnitude higher than the usual UK National Institute for Health and Care Excellence threshold used for other diseases[8]. Delayed health interventions for serious conditions, although difficult to mitigate, represent a further life loss burden.

[5]Based on Office for National Statistics (2019) lifetables, ONS COVID-19 fatality by age data, a mid range IFR of 0.006 and a 1 year lower bound life expectancy adjustment for co-morbidities based on Hanlon et al. (2020). Given reported ICU survival rates, severe ICU overload could plausibly double the burden.

[6]The life expectancy gap between those in the upper and lower half of the UK income scale grew by 14 weeks in the aftermath of 2008, a loss of life that is difficult to attribute to confounders. See Marmot et al. (2020) especially figure 2.5.

[7]Marmot et al. (2020) figure 2.3 suggests 140-240 million life years, depending on the reference quantile (e.g. 0.5 or 0.9).

[8]The central scenario of the UK Office for Budget Responsibility July 14th report has excess borrowing peaking at £660 billion, suggesting a cost per life year in excess of £250 thousand. The NICE threshold is £20-30 thousand.

Acknowledgements

I am very grateful to Robert Verity for his unsolicited offer to help get access to the duration data for England, and for making the effort to get the permissions and follow through on this. I am also grateful to an anonymous referee for some useful suggestions and for comments that inspired figure 4.

Methods

Direct inference about (1) uses the empirical Bayes approach of Wood et al. (2016) in which the smooth functions are estimated by penalized likelihood maximisation (e.g. Green and Silverman, 1994), with the smoothing parameters and θ estimated by Laplace approximate marginal likelihood maximization. Writing $\boldsymbol{\beta}$ for the combined vector of basis coefficients for f and f_w, the penalized version of the log likelihood, $l(\boldsymbol{\beta})$, can be written

$$l(\boldsymbol{\beta}) - \frac{\lambda_f}{2}\int f^{[2]}(t)^2 dt - \frac{\lambda_w}{2}\int f_w^{[2]}(d)^2 dd = l(\boldsymbol{\beta}) - \frac{1}{2}\boldsymbol{\beta}^\mathsf{T}\mathbf{S}_\lambda\boldsymbol{\beta}$$

where $\mathbf{S}_\lambda = \lambda_f\mathbf{S}_f + \lambda_w\mathbf{S}_w$: $\mathbf{S}_f$ and $\mathbf{S}_w$ are known constant positive semi-definite matrices. Smoothing parameters, λ_f and λ_w, control the smoothness of f and f_w. Let $\hat{\boldsymbol{\beta}}$ be the maximizer of the penalized log likelihood, and $\mathbf{H}$ its negative Hessian at $\hat{\boldsymbol{\beta}}$. Viewing the penalty as being induced by an improper Gaussian prior, $\boldsymbol{\beta} \sim N(\mathbf{0}, \mathbf{S}_\lambda^-)$, $\hat{\boldsymbol{\beta}}$ is also the MAP estimate of $\boldsymbol{\beta}$. Furthermore in the large sample limit

$$\boldsymbol{\beta}|\mathbf{y} \sim N(\hat{\boldsymbol{\beta}}, (\mathbf{H}+\mathbf{S}_\lambda)^{-1}). \tag{2}$$

Writing the density in (2) as π_g, and the joint density of $\mathbf{y}$ and $\boldsymbol{\beta}$ as $\pi(\mathbf{y}, \boldsymbol{\beta})$, the Laplace approximation to the marginal likelihood for the smoothing parameters $\boldsymbol{\lambda}$ and θ is $\pi(\boldsymbol{\lambda}, \theta) = \pi(\mathbf{y}, \boldsymbol{\beta})/\pi_g(\boldsymbol{\beta}|\mathbf{y})$. Nested Newton iterations are used to find the values of $\log(\boldsymbol{\lambda}), \theta$ maximizing $\pi(\boldsymbol{\lambda}, \theta)$ and the corresponding $\hat{\boldsymbol{\beta}}$ (for details see Wood et al., 2016).

Given (2) credible intervals for f are readily computed, but it is also straightforward to make inferences about when the peak in f occurs. Simply simulate replicate coefficient vectors from (2) and find the day of occurrence of the peak for each corresponding underlying death rate function, f.

A fully Bayesian approach was taken for inference about f_c, via both an efficient hybrid of the empirical Bayes approach with Metropolis Hastings sampling, and a more expensive Gibbs sampling approach. Both are described in the next subsection.

Fast estimation, renewal model and Gibbs sampling

The model formulated in terms of f_c can also be estimated using the framework of Wood et al. (2016). To do this requires expressions for the negative binomial deviance (or log likelihood) and its derivative vector and Hessian matrix w.r.t. the model coefficients. The non-standard structure of the model means that these must be worked out explicitly, rather than relying on standard software.

First consider the negative binomial deviance for observation i,

$$D_i = 2y_i \log\{\max(1, y_i)/\mu_i\} - (y_i+\theta)\log\{(y_i+\theta)/(\mu_i+\theta)\},$$

$$\frac{\mathrm{d}D_i}{\mathrm{d}\mu_i} = 2\left(\frac{y_i+\theta}{\mu_i+\theta} - \frac{y_i}{\mu_i}\right) \quad \text{and} \quad \frac{\mathrm{d}^2 D_i}{\mathrm{d}\mu_i^2} = 2\left(\frac{y_i}{\mu_i^2} - \frac{y_i+\theta}{(\mu_i+\theta)^2}\right).$$

These need to be transformed into derivatives w.r.t. $\boldsymbol{\beta}$, as follows

$$\frac{\partial D_i}{\partial \beta_j} = \frac{\mathrm{d}D_i}{\mathrm{d}\mu_i}\frac{\partial \mu_i}{\partial \beta_j} \quad \text{and} \quad \frac{\partial^2 D_i}{\partial \beta_j \partial \beta_k} = \frac{\mathrm{d}^2 D_i}{\mathrm{d}\mu_i^2}\frac{\partial \mu_i}{\partial \beta_j}\frac{\partial \mu_i}{\partial \beta_k} + \frac{\mathrm{d}D_i}{\mathrm{d}\mu_i}\frac{\partial^2 \mu_i}{\partial \beta_j \partial \beta_k}.$$

Writing $\mathbf{X}^f$ and $\mathbf{X}^w$ for the model matrices for the smooth terms $\log f_c$ and f_w, we have $\boldsymbol{\delta} = \mathbf{B}\mathbf{f}_c$ where $\mathbf{f}_c = \exp(\mathbf{X}^f\boldsymbol{\beta}^f)$ (here $\exp(\cdot)$, division and multiplication are applied element-wise to vectors), and $\mathbf{f}_w = \mathbf{X}^w\boldsymbol{\beta}^w$. Then $\boldsymbol{\mu} = \exp(\log\boldsymbol{\delta} + \mathbf{f}_w)$, while

$$\frac{\partial\boldsymbol{\mu}}{\partial\boldsymbol{\beta}^f} = \text{diag}(\boldsymbol{\mu}/\boldsymbol{\delta})\mathbf{B}\frac{\partial\mathbf{f}_c}{\partial\boldsymbol{\beta}^f}, \quad \frac{\partial\boldsymbol{\mu}}{\partial\boldsymbol{\beta}^w} = \text{diag}(\boldsymbol{\mu})\mathbf{X}^w, \quad \frac{\partial^2\boldsymbol{\mu}}{\partial\beta^w_j\partial\beta^w_k} = \boldsymbol{\mu}\mathbf{X}^w_{\cdot,j}\mathbf{X}^w_{\cdot,k},$$

$$\frac{\partial^2\boldsymbol{\mu}}{\partial\beta^f_j\partial\beta^f_k} = \text{diag}(\boldsymbol{\mu}/\boldsymbol{\delta})\mathbf{B}\frac{\partial^2\mathbf{f}_c}{\partial\beta^f_j\partial\beta^f_k} \text{ and } \frac{\partial^2\boldsymbol{\mu}}{\partial\beta^f_j\partial\beta^w_k} = \text{diag}(\mathbf{X}^w_{\cdot,k}\boldsymbol{\mu}/\boldsymbol{\delta})\mathbf{B}\frac{\partial\mathbf{f}_c}{\partial\boldsymbol{\beta}^f}.$$

For the given representation of $\mathbf{f}_c$

$$\frac{\partial\mathbf{f}_c}{\partial\boldsymbol{\beta}^f} = \text{diag}(\mathbf{f}_c)\mathbf{X}^f \text{ and } \frac{\partial^2\mathbf{f}_c}{\partial\beta^f_j\partial\beta^f_k} = \text{diag}(\mathbf{f}_c)\mathbf{X}^f_{\cdot,j}\mathbf{X}^f_{\cdot,k}$$

Alternatively $\mathbf{f}_c$ can be produced from the epidemic model of Flaxman et al. (2020), with $\log R_t$ in that model represented as a spline of time, and the initial innoculum also a free parameter. In the notation of Flaxman *et al.* the number of infections each day are denoted c_t. Given an initial c_1 the model is iterated from $t = 2$ as follows

$$c_t = \left(1 - \sum_{i=1}^{t-1} C_i/N\right) R_t \sum_{\tau=1}^{t-1} c_\tau g_{t-\tau} \tag{3}$$

where N is the total initially susceptible population, $g_1 = \int_0^{1.5}\gamma(x)dx$ and $g_j = \int_{j-.5}^{j+.5}\gamma(x)dx$ for $j > 1$. γ is the p.d.f. of a Gamma distribution with shape parameter $0.5/0.62^2$ and scale parameter 0.62^{-2}. The c_t values multiplied by the assumed infection fatality rate give $\mathbf{f}_c$. The level of the IFR only matters for the damping term in the first bracket of the expression for c_t — this has almost no effect in practice. $\log R_t$ is represented using a spline basis, with associated penalty as for the other models, while $\log c_1$ is also treated as a free parameter. Routine application of the chain rule to (3) then gives the corresponding iterations for the derivatives of c_t (and hence $\mathbf{f}_c$) w.r.t the spline coefficients and $\log c_1$.

Given these expressions and the penalties, $\hat{\boldsymbol{\beta}}$ can be obtained by Newton iteration, given smoothing parameters. To estimate smoothing parameters, the simplest approach is to use Wood and Fasiolo (2017), alternating generalized Fellner Schall updates of the smoothing parameters with updates of $\hat{\boldsymbol{\beta}}$ given those smoothing parameters. The negative binomial θ is fixed at its estimate from model (1). This finds the smoothing parameters to approximately maximise the model marginal likelihood. The non-linearity of the renewal equation model means that some effort is required to get non-absurd starting values. I got these by a few minutes of experimentation with simple step functions for the initial $\log R_t$ to get death trajectories of roughly the shape and amplitude of the true trajectories (this does not have to be very accurate — my initial fit deviances were a couple of orders of magnitude greater than for the final fit).

Given θ and the smoothing parameters, the approximate posterior (2) can be used directly, or as the basis for the proposal distribution in a simple Metropolis Hastings sampler. A fairly efficient sampler results from alternating fixed proposals based on (2) with random walk proposals based on zero mean Gaussian steps with a shrunken version of the posterior covariance matrix. By this method, effective sample sizes > 5000 for each coefficient took about 40 seconds computing on a low specification laptop.

The model can also be implemented using the JAGS software for Gibbs sampling (Plummer, 2003; Plummer et al., 2006), making use of the automatic code template generation described in Wood (2016) for reliable implementation of spline smoothers in JAGS. 5×10^6 samples were generated, retaining every 500th sample. This was sufficient to ensure effective sample sizes in the hundreds for even the slowest mixing parameters, while most parameters had effective sample sizes close to 10000. Trace plots suggested rapid convergence. This approach does not fix the smoothing parameters or θ at fixed estimates, but the results are very similar to those obtained by the previous method at the cost of several hours of computing time on the same hardware.

References

Flaxman, S., S. Mishra, A. Gandy, H. J. T. Unwin, T. A. Mellan, H. Coupland, C. Whittaker, H. Zhu, T. Berah, J. W. Eaton, et al. (2020). Estimating the effects of non-pharmaceutical interventions on COVID-19 in Europe. *Nature 584*(7820), 257–261.

Green, P. J. and B. W. Silverman (1994). *Nonparametric Regression and Generalized Linear Models*. Chapman & Hall.

Hanlon, P., F. Chadwick, A. Shah, R. Wood, J. Minton, G. McCartney, C. Fischbacher, F. S. Mair, D. Husmeier, and J. Matthiopoulos (2020). COVID-19 exploring the implications of long-term condition type and extent of multimorbidity on years of life lost: a modelling study. https://wellcomeopenresearch.org/articles/5-75.

Huang, C., Y. Wang, X. Li, L. Ren, J. Zhao, Y. Hu, L. Zhang, G. Fan, J. Xu, X. Gu, et al. (2020). Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China. *The Lancet 395*(10223), 497–506.

Lauer, S. A., K. H. Grantz, Q. Bi, F. K. Jones, Q. Zheng, H. R. Meredith, A. S. Azman, N. G. Reich, and J. Lessler (2020). The incubation period of coronavirus disease 2019 (COVID-19) from publicly reported confirmed cases: estimation and application. *Annals of internal medicine*.

Linton, N. M., T. Kobayashi, Y. Yang, K. Hayashi, A. R. Akhmetzhanov, S.-m. Jung, B. Yuan, R. Kinoshita, and H. Nishiura (2020). Incubation period and other epidemiological characteristics of 2019 novel coronavirus infections with right truncation: a statistical analysis of publicly available case data. *Journal of clinical medicine 9*(2), 538.

Marmot, M., J. Allen, T. Boyce, P. Goldblatt, and J. Morrison (2020). *Health Equity in England: The Marmot Review 10 Years On*. The Health Foundation.

McAloon, C. G., A. Collins, K. Hunt, A. Barber, A. Byrne, F. Butler, M. Casey, J. M. Griffin, E. Lane, D. McEvoy, P. Wall, M. J. Green, L. O'Grady, and S. J. More (2020). The incubation period of COVID-19: A rapid systematic review and meta-analysis of observational research. *medRXiv*.

Office for National Statistics (2019). National life tables, UK: 2016 to 2018. https://www.ons.gov.uk/peoplepopulationandcommunity/birthsdeathsandmarriages/lifeexpectancies/bulletins/nationallifetablesunitedkingdom/2016to2018.

Office for National Statistics (2020, May). Deaths registered weekly in England and Wales. https://www.ons.gov.uk/peoplepopulationandcommunity/birthsdeathsandmarriages/deaths/bulletins/deathsregisteredweeklyinenglandandwalesprovisional/latest.

Plummer, M. (2003). JAGS: A program for analysis of Bayesian graphical models using Gibbs sampling. In *Proceedings of the 3rd International Workshop on Distributed Statistical Computing (DSC 2003).*, pp. 20–22.

Plummer, M., N. Best, K. Cowles, and K. Vines (2006). coda: Convergence diagnosis and output analysis for MCMC. *R News 6*(1), 7–11.

Tay, M. Z., C. M. Poh, L. Rénia, P. A. MacAry, and L. F. Ng (2020). The trinity of COVID-19: immunity, inflammation and intervention. *Nature Reviews Immunology*, 1–12.

Verity, R., L. C. Okell, I. Dorigatti, P. Winskill, C. Whittaker, N. Imai, G. Cuomo-Dannenburg, H. Thompson, P. G. Walker, H. Fu, et al. (2020). Estimates of the severity of coronavirus disease 2019: a model-based analysis. *The Lancet Infectious Diseases*.

Wang, D., B. Hu, C. Hu, F. Zhu, X. Liu, J. Zhang, B. Wang, H. Xiang, Z. Cheng, Y. Xiong, et al. (2020). Clinical characteristics of 138 hospitalized patients with 2019 novel coronavirus–infected pneumonia in Wuhan, China. *Jama 323*(11), 1061–1069.

Wood, S. N. (2016). Just another Gibbs additive modeller: Interfacing JAGS and mgcv. *Journal of Statistical Software 75*(7).

Wood, S. N. and M. Fasiolo (2017). A generalized Fellner-Schall method for smoothing parameter optimization with application to Tweedie location, scale and shape models. *Biometrics 73*(4), 1071–1081.

Wood, S. N., N. Pya, and B. Säfken (2016). Smoothing parameter and model selection for general smooth models (with discussion). *Journal of the American Statistical Association 111*, 1548–1575.

Wood, S. N., E. C. Wit, M. Fasiolo, and P. J. Green (2020). COVID-19 and the difficulty of inferring epidemiological parameters from clinical data. *The Lancet Infectious Diseases*.

Worldometer (2020). https://www.worldometers.info/coronavirus/.

Wu, J. T., K. Leung, M. Bushman, N. Kishore, R. Niehus, P. M. de Salazar, B. J. Cowling, M. Lipsitch, and G. M. Leung (2020). Estimating clinical severity of COVID-19 from the transmission dynamics in Wuhan, China. *Nature Medicine 26*(4), 506–510.

Zhou, F., T. Yu, R. Du, G. Fan, Y. Liu, Z. Liu, J. Xiang, Y. Wang, B. Song, X. Gu, et al. (2020). Clinical course and risk factors for mortality of adult inpatients with COVID-19 in Wuhan, China: a retrospective cohort study. *The Lancet*.

Supplementary code and data

The code and data used in the paper are supplied here. The conditions under which the CHESS data were supplied mean that I can not make them available.

```
## functions for post fit MCMC...

lpi <- function(beta,y,Xf,Xw,B,St,theta,Dev=Dev0) {
## log likelihood + prior (excluding fixed constants)
  d <- Dev(beta,y,Xf,Xw,B,theta=theta,deriv=FALSE)
  -drop(d$D +  beta %*% St %*% beta)/2
}

rmvt <- function(n,mu,V,df) {
## simulate multivariate t variates
  y <- rmvn(n,mu*0,V)
  v <- rchisq(n,df=df)
  t(mu + t(sqrt(df/v)*y))
}

dmvt <- function(x,mu,V,df,R=NULL) {
## multivariate t log density...
  p <- length(mu);
  if (is.null(R)) R <- chol(V)
  z <- forwardsolve(t(R),x-mu)
  k <- - sum(log(diag(R))) - p*log(df*pi)/2 + lgamma((df+p)/2) - lgamma(df/2)
  k - if (is.matrix(z)) (df+p)*log1p(colSums(z^2)/df)/2 else (df+p)*log1p(sum(z^2)/df)/2
}

dmvn <- function(x,mu,V,R=NULL) {
## multivariate normal density mgcv:::rmvn can be used for generation
  if (is.null(R)) R <- chol(V)
  z <- forwardsolve(t(R),x-mu)
  -colSums(z^2)/2-sum(log(diag(R))) - log(2*pi)*length(mu)/2
}

## Flaxman et al (2020) renewal model functions...

renew <- function(X,beta,N=6.7e7,ifr=.006,deriv=TRUE) {
## simple renewal model with sensitivities...
## Gamma dist used has mean 6.5 and CV 0.62. i.e. shape = 6.5 * .62^2, scale = 1/.62^2.
## beta[1] = log(z[1]) - the innoculum
## R = exp(X%*%beta[-1])
## z is predicted daily infections
  nt <- nrow(X)
  R <- exp(X %*% beta)
  i0 <- exp(beta[1]) ## initial infection
  z <- rep(i0,nt)    ## infections
  p <- length(beta)
  d1z <- matrix(0,p,nt)      ## first deriv infections w.r.t. coefs
  d2z <- array(0,c(p,p,nt))  ## second deriv infections w.r.t. coefs
  d1z[1,1] <- d2z[1,1,1] <- i0 ## deriv z[1] w.r.t. beta[1]
  ## set up renewal model...
  t <- 0:nt+.5; t[1] <- 0
  g <- pgamma(t[-1],shape=6.5*.62^2,scale=1/.62^2) - pgamma(t[-nt],shape=6.5*.62^2,scale=1/.62^2)
  for (t in 2:nt) { ## main iteration
    damp <- (1-sum(z[1:(t-1)])/N)      ## damping as susceptible fraction declines
    rnew <- sum(z[1:(t-1)]*g[(t-1):1]) ## the renewal process itself
    z[t] <- damp*R[t]*rnew             ## new infections
    ## what follows is just the chain rule applied to the preceding terms...
    if (deriv) {
      d1damp <- -rowSums(d1z[,1:(t-1),drop=FALSE])/N
      d1R <- R[t]*X[t,]
      d1rnew <- colSums(t(d1z[,1:(t-1),drop=FALSE])*g[(t-1):1])
      d1z[,t] <- d1damp*R[t]*rnew + damp*d1R*rnew + damp*R[t]*d1rnew
      d2damp <- -apply(d2z[,,1:(t-1),drop=FALSE],c(1,2),sum)/N
      d2R <- R[t] * outer(X[t,],X[t,])
      d2rnew <- apply(d2z[,,1:(t-1),drop=FALSE],c(1,2),function(x) sum(x*g[(t-1):1]))
      d2z[,,t] <- d2damp*R[t]*rnew + outer(d1damp,d1R)*rnew + outer(d1damp,d1rnew)*R[t] +
                  outer(d1R,d1damp)*rnew + damp*d2R*rnew + damp*outer(d1R,d1rnew) +
outer(d1rnew,d1damp)*R[t] + damp * outer(d1rnew,d1R) + damp*R[t]*d2rnew
    } ## if deriv
  }
  ## multiply by ifr to get fatal infection profile...
  list(z=z*ifr,d1z=d1z*ifr,d2z=d2z*ifr)
} ## renew

Devr <- function(beta,y,Xf,Xw,B,theta=30,deriv=TRUE) {
## negative binomial deviance, grad and Hessian
## for the renewal model in which Rt is the smooth function
## controlling the generation of infections
## beta - model cofficients
## y - response (deaths)
## Xf model matrix for log Rt. First column should be zero
## to allow first parameter of this model to be used as
## log innoculum.
## Xw model matrix for weekly cycle in deaths/reported deaths
## B the forward matrix mapping infections to death rate
## theta - negative binomial theta
  ind <- 1:ncol(Xf)
  betaf <- beta[ind]  ## controls incidence
```

```
  betaw <- beta[-ind] ## weekly cycle
  r0 <- renew(Xf,betaf,deriv=deriv) ## run the renewal model and its sensitivities
  delta <- drop(B %*% r0$z) ## expected deaths

  etaw <- drop(Xw %*% betaw)
  mu <- exp(log(delta)+etaw)

  muth <- mu + theta
  yth <- y + theta
  dev <-  2* sum(y * log(pmax(1, y)/mu) - yth * log(yth/muth))
  if (!deriv) return(list(D=dev))

  delta1 <- B %*% t(r0$d1z)
  mu1 <- cbind(mu/delta*delta1,mu*Xw) ## dmu/dbeta
  D1 <- colSums(2*(yth/muth-y/mu)*mu1) ## dD/dbeta
  ## Hessian...
  p <- length(beta)
  pf <- ncol(Xf)
  D2 <- matrix(0,p,p)
  for (i in 1:p) {
    if (i<=pf) {
      mu2 <-  cbind((mu/delta)*(B %*% t(r0$d2z[i,,])), drop(B %*% r0$d1z[i,])*mu/delta*Xw)
    } else {
      ii <- i - pf
      mu2 <- cbind((B%*%t(r0$d1z))*mu/delta*Xw[,ii],(mu*Xw[,ii])*Xw)
    }
    for (j in 1:i) {
      D2[i,j] <- D2[j,i] <- sum(-2 * (yth/muth^2 - y/mu^2)*mu1[,i]*mu1[,j] +
            2*(yth/muth-y/mu)*mu2[,j])
    }
  }
  list(D=dev,D1=D1,D2=D2,mu=mu,f=r0$z,fw=etaw)
} ## Devr

plotRt <- function(res,last.day=NULL,ylim=NULL,ylab="log(R)",times=c(-9,0,3,7)) {
## plot CI for R computed using renewal model
  ii <- 1:ncol(res$Xf)
  lR <- drop(res$Xf %*% res$beta[ii])
  seR <- rowSums(res$Xf*(res$Xf %*% res$Vb[ii,ii]))^.5
  n <- length(lR); day <- 1:n-21
  Rq <- rbind(lR+2*seR,lR+seR,lR,lR-seR,lR-2*seR)
  #Rq <- log(apply(R,1,quantile,prob=c(.025,.16,.5,.84,.975)))
  #Rq[,1:5] <- NA
  xlim <- if (is.null(last.day)) range(day) else c(min(day),last.day)
  if (is.null(ylim)) ylim <- range(Rq)
  c1 <- 1.3
  plot(day,Rq[3,],type="l",ylim=ylim,xlim=xlim,ylab=ylab,cex.lab=c1)
  polygon(c(day,day[n:1]),c(Rq[1,],Rq[5,n:1]),col="lightgrey",border=NA)
  polygon(c(day,day[n:1]),c(Rq[2,],Rq[4,n:1]),col="grey",border=NA)
  lines(day,Rq[3,]);abline(v=11,col=2);
  if (!is.null(times)) for (i in 1:length(times)) abline(v=times[i],lty=i+1)
  abline(0,0,col=4)
} ## plotRt

## Main 'standard' model functions...

Dev0 <- function(beta,y,Xf,Xw,B,theta=30,deriv=TRUE) {
## negative binomial deviance, grad and Hessian
## beta - model cofficients
## y - response (deaths)
## Xf model matrix for smooth underlying infection rate
## Xw model matrix for weekly cycle in deaths/reported deaths
## B the forward matrix mapping infections to death rate
## theta - negative binomial theta
  ind <- 1:ncol(Xf)
  betaf <- beta[ind]
  betaw <- beta[-ind]
  eta <- drop(Xf %*% betaf)
  f <- exp(eta)
  muf <- drop(B %*% f)
  muf1 <- B %*% (f*Xf)
  etaf1 <- muf1/muf ## detaf/dbeta
  etaw <- drop(Xw %*% betaw)
  mu <- exp(log(muf)+etaw)
  mu1 <- cbind(mu*etaf1,mu*Xw) ## dmu/dbeta
  muth <- mu + theta
  yth <- y + theta
  dev <-  2* sum(y * log(pmax(1, y)/mu) - yth * log(yth/muth))
  if (!deriv) return(list(D=dev))
  D1 <- colSums(2*(yth/muth-y/mu)*mu1) ## dD/dbeta
  ## Hessian...
  p <- length(beta)
  pf <- ncol(Xf)
  D2 <- matrix(0,p,p)
  for (i in 1:p) {
    if (i<=pf) {
      mu2 <-  cbind((mu/muf)*(B %*% (f*Xf*Xf[,i])), etaf1[,i]*mu*Xw)
    } else {
      ii <- i - pf
      mu2 <- cbind(etaf1*mu*Xw[,ii],(mu*Xw[,ii])*Xw)
    }
    for (j in 1:i) {
      D2[i,j] <- D2[j,i] <- sum(-2 * (yth/muth^2 - y/mu^2)*mu1[,i]*mu1[,j] +
```

```
            2*(yth/muth-y/mu)*mu2[,j])
    }
  }
  list(D=dev,D1=D1,D2=D2,mu=mu,f=f,fw=etaw)
} ## Dev0

fit1 <- function(beta,sp,y,Xf,Xw,B,S,theta=30,tol=1e-7,Dev=Dev0) {
## fit smooth model by Newton iteration, given smoothing parameters...
  d <- Dev(beta,y,Xf,Xw,B,theta=theta)
  p <-  length(beta)
  St <- matrix(0,p,p)
  ii <- 1:ncol(Xf);St[ii,ii] <- sp[1]*S[[1]]
  ii <- ncol(Xf)+1:ncol(Xw);St[ii,ii] <- sp[2]*S[[2]]
  pdev <- drop(d$D +  beta %*% St %*% beta)
  nok <- TRUE
  while (nok) {
    eh <- eigen(d$D2 + 2* St) ## eigen decomp of penalized Hessian
    iv <- eh$values;thresh <- if (min(iv)<0) max(iv)*1e-5 else  max(iv)*1e-10
    iv[iv<thresh] <- thresh
    iv <- 1/iv   ## perturb to +ve def (guarantee descent direction)
    gr <- drop(d$D1 + St%*%beta*2) ## add smoothing penalty/prior grad to dev grad
    if (all(abs(gr)<pdev*tol)) break ## converged
    step <- - drop(eh$vectors %*% (iv*(t(eh$vectors) %*% gr))) ## Newton step
    pdev1 <- 2*pdev
    while (!is.finite(pdev1)||pdev1>pdev+1e-12) { ## Newton step control loop
      beta1 <- beta + step
      d1 <- Dev(beta1,y,Xf,Xw,B,theta=theta)
      pdev1 <- drop(d1$D + beta1 %*% St %*% beta1)
      if (!is.finite(pdev1)||pdev1>pdev) step <- step/ if (is.finite(pdev1)) 2 else 100
      step.fail <- all.equal(beta+step,beta,tolerance=.Machine$double.eps^.75)==TRUE
      if (step.fail) break
    }
    if (step.fail) break
    d <- d1;pdev <- pdev1;beta <- beta1
  }
  list(beta=beta,Vb=solve(d$D2/2 + St),step.fail=step.fail)
} ## fit1


full.fit <- function(deaths,day,dow,theta,dilation=0,mcmc=TRUE,ei2d=3.19,si2d=.44,renew=FALSE) {
## fit model - dilation 0 for none, 4 for as paper.
## log(i2d) ~ N(ei2d,si2d^2)
## NB theta must be supplied here - usually from simple smooth additive fit to deaths
  day1 <- day;
  dday <- 10
  ii <- day>dday-1;day1[ii] <- day1[ii]+dilation/2
  ii <- day>dday;day1[ii] <- day1[ii]+dilation
  ii <- day>dday+1;day1[ii] <- day1[ii]+dilation/2

  sm <- smoothCon(s(day,k=20,m=2),data=data.frame(day=day1))[[1]]
  eps <- 1e-4
  Xg <- PredictMat(sm,data.frame(day=day1+eps))
  smw <- smoothCon(s(dow,k=7,bs="cc"),data=data.frame(dow=dow),
                absorb.cons=TRUE,knots=list(dow=c(0,7)))[[1]]

  Xf <- sm$X;Xw <- smw$X;
  Xg <- (Xg-sm$X)/eps ## the infection smooth grad...
  S <- list(sm$S[[1]],smw$S[[1]])
  if (renew) { ## then use simple epidemic model
    Xf <- cbind(0,Xf) ## expand to allow initial condition estimation
    S[[1]] <- rbind(0,cbind(0,S[[1]])) ## pad S[[1]] accordingly
    Dev <- Devr ## use alternative Dev
    yy <- c(rep(1,30),rep(-.3,nrow(Xf)-30))
    beta <- rep(0,ncol(Xf)+ncol(Xw));
    bb <- coef(lm(yy~Xf[,-1]-1))
    beta[1+1:length(bb)] <- bb
    beta[1] <- 8.5
  } else {
    beta <- rep(0,ncol(Xf)+ncol(Xw))
    Dev=Dev0
  }
  nc <- length(deaths)
  ## Set up matrix mapping infections to future deaths based on published
  ## incubation and disease duration distributions...
  d <- dlnorm(1:nc,ei2d,si2d)
  B <- matrix(0,nc,nc)
  for (i in 1:nc) { ## map case rate day i-1 to death rate day i ...
    B[,i] <- c(rep(0,i-1),d[1:(nc-i+1)])
  }

  ## Empirical Bayes model estimation. NB theta as supplied, smoothing parameters to
  ## maximize approximate Laplace approximate Marginal Likelihood by Extended
  ## Fellner Schall algorithm (Wood and Fasiolo (2017) Biometrics)...


  lambda <- c(1,1);rank <- c(sm$rank,smw$rank)
  iif <- 1:ncol(Xf);iiw <- ncol(Xf) + 1:ncol(Xw)
  for (i in 1:10) {
    ## run Newton for fit given smoothing params, lambda...
    f <- fit1(beta,lambda,deaths,Xf,Xw,B,S,theta=theta,tol=1e-7,Dev=Dev)
    beta <- f$beta;Vb <- f$Vb;
    d <- Dev(beta,deaths,Xf,Xw,B,theta=theta) ## extract best fit deviance + derivs
    if (i==1) D0 <- d$D else {
```

```
      if (abs(D0-d$D)<1e-3*d$D) break ## stop when fit change small
      D0 <- d$D
    }
    ## Commented out forces Hessian of Deviance to be +ve def, not just
    ## Hessian of penalized deviance. Only needed if lambda iteration
    ## diverges to negative
    #ed <- eigen(d$D2/2)
    #ed$values[ed$values<0] <- 0
    #D2 <- ed$vectors%*%(ed$values*t(ed$vectors))
    #Vb <- solve(D2+lambda*sm$S[[1]])

    ## update smoothing parameters, lambda...
    lambda[1] <- drop(lambda[1]*(rank[1]/lambda[1] - sum(Vb[iif,iif]*S[[1]]))/
                 (beta[iif]%*%S[[1]]%*%beta[iif]))
    lambda[2] <- drop(lambda[2]*(rank[2]/lambda[2] - sum(Vb[iiw,iiw]*S[[2]]))/
                 (beta[iiw]%*%S[[2]]%*%beta[iiw]))
  } ## update loop

  ret <- list(Xf=Xf,Xw=Xw,Xg=Xg,S=S,beta=beta,Vb=Vb,B=B,deaths=deaths,theta=theta,
              ei2d=ei2d,si2d=si2d,lambda=lambda)

  if (mcmc) { ## Metropolis Hastings simulation to improve posterior inference
    ## total penalty matrix
    p <- length(beta);St <- matrix(0,p,p)
    ii <- 1:ncol(Xf);St[ii,ii] <- lambda[1]*S[[1]]
    ii <- ncol(Xf)+1:ncol(Xw);St[ii,ii] <- lambda[2]*S[[2]]

    ns <- 100000; t.df <- 4
    bp <- rmvt(ns,beta,Vb,df=t.df) ## beta proposals
    lfp <- dmvt(t(bp),beta,Vb,df=t.df) ## log proposal density

    R <- chol(Vb)
    step <- rmvn(ns,beta*0,Vb/4) ## random walk steps (mgcv::rmvn)

    u <- runif(ns);us <- runif(ns) ## for acceptance check

    bs <- bp;j <- 1;accept <- 0
    lpi0 <- lpi(bs[1,],deaths,Xf,Xw,B,St,theta,Dev=Dev)
    for (i in 2:ns) { ## MH loop
      ## first a static proposal...
      lpi1 <- lpi(bs[i,],deaths,Xf,Xw,B,St,theta,Dev=Dev)
      if (u[i] < exp(lfp[j]-lfp[i]+lpi1-lpi0)) {
        lpi0 <- lpi1;accept <- accept + 1
        j <- i ## row of bs containing last accepted beta
      } else bs[i,] <- bs[i-1,]
      ## now a random walk proposal...
      lpi1 <- lpi(bs[i,]+step[i,],deaths,Xf,Xw,B,St,theta,Dev=Dev)
      if (us[i] < exp(lpi1-lpi0)) { ## accept random walk step
        lpi0 <- lpi1;j <- i
        bs[i,] <- bs[i,] + step[i,]
        lfp[i] <- dmvt(bs[i,],beta,Vb,df=4,R=R) ## have to update static proposal density
      }
      if (i%%2000==0) cat(".")
    }
    accept <- accept/ns
    ii <- 1:ncol(Xf); fi <- Xf%*%t(bs[,ii])
    fsim <- exp(apply(fi,1,quantile,probs=c(.025,.16,.5,.84,.975)))
    ret$accept <- accept;ret$fsim <- fsim;ret$bs <- bs
  } ## if (mcmc)
  ret
} ## full.fit



## Now for plotting the results...

plot.ip <- function(res,mcmc=TRUE,approx=TRUE,last.day=NULL,lock.down=11,ylab="fatal infections",c1=1,renew=FALSE) {
## infection profile plotting
  if (is.null(res$fsim)) mcmc <- FALSE
  iif <- 1:ncol(res$Xf)
  se <- rowSums((res$Xf %*% res$Vb[iif,iif])*res$Xf)^.5
  if (renew) Dev <- Devr
  d <- Dev(res$beta,res$deaths,res$Xf,res$Xw,res$B,theta=res$theta)
  lag <- 21 ## get day zero timing right at 13th March
  day <- 1:length(res$deaths)-lag
  xlim <- if (is.null(last.day)) range(day) else c(min(day),last.day)
  ll2 <- exp(log(d$f)-2*se);ul2 <- exp(log(d$f)+2*se)
  n <- length(ll2)
  ll <- exp(log(d$f)-se);ul <- exp(log(d$f)+se)
  ii <- 1:(length(d$f)-23);yl<- max(ul2[ii])
  plot(day,d$f,type="l",ylim=c(0,yl),ylab=ylab,xlim=xlim,cex.lab=c1)
  ll <- exp(log(d$f)-se);ul <- exp(log(d$f)+se)
  if (mcmc) {
    polygon(c(day,day[n:1]),c(res$fsim[1,],res$fsim[5,n:1]),col="lightgrey",border=NA)
    polygon(c(day,day[n:1]),c(res$fsim[2,],res$fsim[4,n:1]),col="grey",border=NA)
    lines(day,res$fsim[3,])
    if (approx) {
      lines(day,ll2,lty=3,col=4);lines(day,ul2,lty=3,col=4)
      lines(day,ll,lty=2,col=4);lines(day,ul,lty=2,col=4)
      lines(day,d$f,col=4)
    }
  } else {
    polygon(c(day,day[n:1]),c(ul2,ll2[n:1]),col="lightgrey",border=NA)
```

```
    polygon(c(day,day[n:1]),c(ul,ll[n:1]),col="grey",border=NA)
    lines(day,d$f)
  }
  ## add the posterior for peak location
  nb <- if (renew) 5000 else 10000
  bb <- rmvn(nb,res$beta[iif],res$Vb[iif,iif])
  if (renew) { ## have to run the model to get infection profiles and their peaks
    fb <- matrix(0,nb,nrow(res$Xf))
    for (i in 1:nb) fb[i,] <- renew(res$Xf,bb[i,],deriv=FALSE)$z
  } else fb <- bb %*% t(res$Xf)
  peak <- apply(fb[,ii],1,function(x) which(x==max(x)))
  pt <- tabulate(peak)
  scale <- 0.1*yl/max(pt)
  for (i in 1:length(pt)) if (pt[i]>4) lines(c(i-lag,i-lag),c(0,pt[i]*scale),lwd=2,col=4)
  tp <- which(pt==max(pt)) - lag
  mtext(tp,side=1,line=0,at=tp,col=4,cex=.7)
  if (is.finite(lock.down)) {
    abline(v=lock.down,col=2) ## 1st day of lockdown
    mtext(lock.down,side=1,line=1,at=lock.down,col=2,cex=.7)
  }
} ## plot.ip


sanity.plot <- function(res,b,ylab="deaths",c1=1,c2=1,Dev=Dev0) {
## Sanity check the posterior mode infection profile.
## b is a simple gam death model fit, res is a full infection profile fit
  d <- Dev(res$beta,res$deaths,res$Xf,res$Xw,res$B,theta=res$theta)
  fi <- round(d$f)
  n <- length(fi)
  n.rep <- 100
  death <- matrix(0,n.rep,n)
  for (j in 1:n.rep) for (i in 1:n) {
    t <- round(rlnorm(fi[i],res$ei2d,res$si2d))
    t <- t[i+t-1<=n] ## discard deaths beyond end of data
    dd <- tabulate(t)
    ii <- 1:length(dd)+i-1
    death[j,ii] <- death[j,ii] + dd
  }
  lag <- 21 ## get day zero timing right at 13th March
  day <- 1:length(res$deaths)-lag
  lag2 <- 0
  plot(day+ lag2,death[1,],type="l",ylim=range(res$deaths),col="grey",xlab="day",ylab=ylab,cex.lab=c1,cex.axis=c2)
  for (j in 2:n.rep) lines(day+lag2,death[j,],col="grey")

  X <- model.matrix(b);X[,21:25] <- 0    ## UK model matrix, weekly removed
  betab <- coef(b); Vbb <- vcov(b); Xj <- X
  fv <- Xj%*%betab                       ## death rates - link scale
  se <- rowSums(Xj*(Xj%*%Vbb))^.5  ## corresponding s.e.
  n <- length(fv)
  ii <- (length(day)-n+1):length(day)
  lines(day[ii],exp(fv))
  lines(day[ii],exp(fv+2*se),lty=2)
  lines(day[ii],exp(fv-2*se),lty=2)
  points(day,res$deaths,col=4)
} ## sanity.plot

r.plot <- function(res,mcmc=TRUE,last.day=NULL,ylab="r",c1=1) {
## 'r' plot...
  day <- 1:length(res$deaths) - 21 #- 1 ## no instant death
  n <- length(day);ii <- 1:ncol(res$Xg)
  if (mcmc) {
    r <- res$Xg %*% t(res$bs[,ii])
    rq <- apply(r,1,quantile,probs=c(.025,.16,.5,.84,.975)) ## get profile CIs
  } else {
    rq <- matrix(0,5,n)
    rq[3,] <- r <- res$Xg %*% res$beta[ii]
    se <- rowSums(res$Xg*(res$Xg%*%res$Vb[ii,ii]))^.5
    rq[1,] <- r - 2*se;rq[2,] <-  r - se
    rq[5,] <- r + 2*se;rq[4,] <-  r + se
  }
  xlim <- if (is.null(last.day)) range(day) else c(min(day),last.day)
  plot(day,rq[3,],type="l",ylim=range(rq),xlim=xlim,ylab=ylab,cex.lab=c1)
  polygon(c(day,day[n:1]),c(rq[1,],rq[5,n:1]),col="lightgrey",border=NA)
  polygon(c(day,day[n:1]),c(rq[2,],rq[4,n:1]),col="grey",border=NA)
  lines(day,rq[3,])
  abline(v=11,col=2) ## 1st day of lockdown
  abline(0,0,col=4)
} ## r.plot


setwd("foo/bar") ## SET WORKING DIRECTORY

## load various daily data - see below for contents...
source("reported-daily.r")

## Extreme simulation. Un-mitigated growth until day before lockdown
## doubling every 3 days, then instant drop and slowly declining.

set.seed(1)
Ey <- 4/1.26
y <- rep(0,length(england)+8)
for (i in 1:31) {
  Ey <- Ey * 1.26; y[i] <- rpois(1,Ey)
```

```
}
Ey <- Ey * .2
for (i in 32:length(y)) {
  y[i] <- rpois(1,Ey); Ey <- Ey *.95
}
n <- length(y);death <- y*0
for (i in 1:n) {
   t <- round(rlnorm(y[i],3.19,.44))
   t <- t[i+t-1<=n] ## discard deaths beyond end of data
   dd <- tabulate(t)
   ii <- 1:length(dd)+i-1
   death[ii] <- death[ii] + dd
}

choose.data <- function(k) {
  ## somewhat sloppy data selecting code...
  data.set <- c("ons","sweden","uk.rep","england","scot","esim")[k]
  cat(data.set,"\n")
  if (data.set=="sweden") { ## Swedish daily reported deaths
    dat <- data.frame(deaths=sweden,day=1:length(sweden),dow = rep(1:7,100)[1:length(sweden)])
    deaths <- c(rep(0,18),sweden)
  } else if (data.set == "scot") { ## scotland - appears to be reported only
    dat <- data.frame(deaths=scot,day=1:length(scot),dow = rep(1:7,100)[1:length(scot)])
    deaths <- c(rep(0,11),scot)
  } else if (data.set=="uk.rep") { ## UK daily reported deaths (old)
    dat <- data.frame(deaths=uk,day=1:length(uk),dow = rep(1:7,100)[1:length(sweden)])
    deaths <- c(rep(0,20),uk)
  } else if (data.set=="england"){ ## NHS hospital deaths England exact day first entry 2 March
    dat <- data.frame(deaths=england,day=1:length(england),dow = rep(1:7,100)[1:length(england)])
    deaths <- c(rep(0,8),england)
  } else if (data.set=="ons") {## ONS E&W daily deaths, all locations, exact date start 5 March
    deaths <- c(rep(0,11),ons)
    dat <- data.frame(deaths=ons,day=1:length(ons),dow = rep(1:7,100)[1:length(ons)])
  } else { ## extreme simulation
    deaths <- death
    dat <- data.frame(day=1:length(england),dow = rep(1:7,100)[1:length(england)])
    dat$deaths <- death[-(1:8)]
  }
  list(deaths=deaths,dat=dat)
}

library(mgcv)

## basic analysis, ONS, NHS, Sweden. Day zero, March 13...

dum <- choose.data(1) ## ONS England
deaths <- dum$deaths; dat <- dum$dat; rm(dum)
nc <- length(deaths);day <- 1:nc-21
dow <- rep(1:7,100)[1:nc] ## day of week
## Basic death rate model, without infection model yet...
bo <- gam(deaths~s(day,k=20)+s(dow,k=7,bs="cc"),family=nb(),data=dat,knots=list(dow=c(0,7)))
theta <- bo$family$getTheta(TRUE) ## Use this negative binomial theta for full model
system.time(reso <- full.fit(deaths,day,dow,theta,dilation=0,mcmc=TRUE,ei2d=3.19,si2d=.44))

dum <- choose.data(4) ## NHS England
deaths <- dum$deaths; dat <- dum$dat; rm(dum)
nc <- length(deaths);day <- 1:nc-21
dow <- rep(1:7,100)[1:nc] ## day of week
## Basic death rate model, without infection model yet...
b <- gam(deaths~s(day,k=20)+s(dow,k=7,bs="cc"),family=nb(),data=dat,knots=list(dow=c(0,7)))
theta <- b$family$getTheta(TRUE) ## Use this negative binomial theta for full model
system.time(res <- full.fit(deaths,day,dow,theta,dilation=0,mcmc=TRUE,ei2d=3.19,si2d=.44))

## Repeat using the Flaxman et al. renewal model and direct inference for R
resn <- full.fit(deaths,day,dow,theta,dilation=0,mcmc=FALSE,ei2d=3.19,si2d=.44,renew=TRUE)
ps <- FALSE
if (ps)  postscript("renewal.eps",width=11,height=3)
par(mfrow=c(1,3),mar=c(5,5,1,1))
plotRt(resn,last.day=70,ylim=c(-2,2))
plot.ip(resn,approx=F,last.day=70,ylab="E fatal infections",c1=1.3,renew=TRUE)
sanity.plot(resn,b,ylab="E hospital deaths",c1=1.3,Dev=Devr)
if (ps) dev.off()


dum <- choose.data(2) ## Sweden
deaths <- dum$deaths; dat <- dum$dat; rm(dum)
nc <- length(deaths);day <- 1:nc-21
dow <- rep(1:7,100)[1:nc] ## day of week
bs <- gam(deaths~s(day,k=20)+s(dow,k=7,bs="cc"),family=nb(),data=dat,knots=list(dow=c(0,7)))
theta <- bs$family$getTheta(TRUE) ## Use this negative binomial theta for full model
res.sw <- full.fit(deaths,day,dow,theta,dilation=0,mcmc=TRUE,ei2d=3.19,si2d=.44)

ps <- FALSE
if (ps) postscript("infections.eps",width=11,height=9)
par(mfrow=c(3,2),mar=c(5,5,1,1));c1= 1.3
plot.ip(reso,approx=F,last.day=70,ylab="EW fatal infections",c1=c1)
sanity.plot(reso,bo,ylab="EW deaths (ONS)",c1=c1)
plot.ip(res,approx=F,last.day=70,ylab="E fatal infections",c1=c1)
sanity.plot(res,b,ylab="E hospital deaths",c1=c1)
plot.ip(res.sw,approx=F,last.day=70,lock.down=NA,ylab="Sweden fatal infections",c1=c1)
r.plot(res,last.day=70,ylab="r England",c1=c1)
if (ps) dev.off()
```

```
## dilation experiment...

dum <- choose.data(4) ## NHS England
deaths <- dum$deaths; dat <- dum$dat; rm(dum)
nc <- length(deaths);day <- 1:nc-21
dow <- rep(1:7,100)[1:nc] ## day of week
bd <- gam(deaths~s(day,k=20)+s(dow,k=7,bs="cc"),family=nb(),data=dat,knots=list(dow=c(0,7)))
theta <- bd$family$getTheta(TRUE) ## Use this negative binomial theta for full model
resd <- full.fit(deaths,day,dow,theta,dilation=5,mcmc=TRUE,ei2d=3.19,si2d=.44)

dum <- choose.data(6) ## Extreme simulation
deaths <- dum$deaths; dat <- dum$dat; rm(dum)
nc <- length(deaths);day <- 1:nc-21
dow <- rep(1:7,100)[1:nc] ## day of week
bed <- gam(deaths~s(day,k=20)+s(dow,k=7,bs="cc"),family=nb(),data=dat,knots=list(dow=c(0,7)))
theta <- bed$family$getTheta(TRUE) ## Use this negative binomial theta for full model
resd.e <- full.fit(deaths,day,dow,theta,dilation=5,mcmc=TRUE,ei2d=3.19,si2d=.44)

if (ps) postscript("dilation.eps",width=11,height=6)
par(mfrow=c(2,2),mar=c(5,5,1,1));c1 <- 1.3
plot.ip(resd.e,approx=F,last.day=70,c1=c1,ylab="simulated fatal infections")
lines(day,y,col=4)
sanity.plot(resd.e,bed,c1=c1,ylab="simulated deaths")
plot.ip(resd,approx=F,last.day=70,c1=c1,ylab="E fatal infections")
sanity.plot(resd,bd,c1=c1,"E hospital deaths")
if (ps) dev.off()
```

The contents of `reported-daily.r`...

```
## sweden from  Folkhlsomyndigheten 4th July, first death is March 11
## https://experience.arcgis.com/experience/09f821667ce64bf7be6f9f87457ed9aa
sweden <- diff(c(0,1,1,2,3,5,7,8,14,21,30,38,49,60,81,103,134,166,201,239,284,332,385,455,534,604,690,780,864,979,1065,1155,1258,1355,1440,
1531,1646,1757,1839,1925,2013,2097,2159,2236,2322,2411,2484,2559,2632,2714,2798,2876,2954,3027,3102,3186,3258,3331,3411,3471,3538,3612,3676,
3737,3787,3833,3890,3938,3991,4052,4091,4145,4198,4253,4309,4352,4394,4422,4461,4501,4541,4580,4625,4665,4701,4727,4772,4810,4839,4872,4910,
4944,4983,5017,5045,5078,5104,5133,5161,5192,5220,5247,5273,5292,5310,5335,5357,5370,5373,5380,5392,5402,5406,5408,5408,5420))

## https://www.england.nhs.uk/statistics/statistical-work-areas/covid-19-daily-deaths/
## first entry is 2nd March (nothing before) NHS exact date retrieved 3 July
england <-c(1,2,0,2,2,0,5,4,1,10,14,20,23,28,40,46,65,63,105,103,149,159,205,264,325,351,359,438,496,574,645,646,697,777,743,726,812,899,790,
739,779,718,698,648,685,639,609,570,522,565,484,501,451,437,385,380,343,341,323,312,306,268,251,259,251,266,255,213,202,195,166,183,162,178,
170,167,137,155,143,153,149,121,128,115,133,138,120,124,116,91,83,94,108,110,83,86,83,80,73,67,76,49,52,42,58,54,57,48,48,38,44,33,41,50,52,
42,33,26)

## www.ons.gov.uk/peoplepopulationandcommunity/birthsdeathsandmarriages/deaths/datasets/weeklyprovisionalfiguresondeathsregisteredinenglandandwales
## ONS E&W 3 July release, death registered up to 27 June. First death is 5th March
ons <- diff(c(0,4,6,6,8,13,15,21,32,47,66,96,143,197,264,338,448,580,754,950,1195,1503,1887,2316,2786,3339,3985,4747,5590,6517,7482,8546,9657,
10718,11905,13251,14477,15679,16923,18187,19313,20445,21596,22786,23948,25017,26029,27062,28023,29005,29941,30847,31655,32442,33192,33916,
34643,35330,36017,36633,37212,37801,38365,38917,39471,39964,40440,40875,41250,41644,42034,42428,42802,43173,43500,43858,44176,44484,44778,
45046,45304,45555,45799,46071,46313,46565,46793,46990,47166,47353,47556,47739,47913,48072,48199,48345,48492,48621))
```

Further code for computing R from the inferred infection profiles.

```
getR <- function(i,gamma=1/3,delta=.2) {
## Compute R implied by incidence/fatal infection profile from simple
## SEIR model. dE/dt = i - gamma*E, dI/dt = gamma E - delta*I
## R = i/(I*delta). Use 1 day discretized version here. Note that
## if all infections are k*i, then k cancels in definition of R.
  n <- length(i)
  E <- 0;I <- i
  for (j in 2:n) {
    E <- i[j-1] + (1-gamma)*E
    I[j] <- gamma*E + (1-delta)*I[j-1]
  }
  return(list(R=i/(I*delta),I=I))
} ## getR

sampleR <- function(res,thin=100,gamma=1/3,delta=1/5) {
## Given samples in res$bs and infection profile model matrix in
## res$Xf compute sample of R profiles from simple SEIR model.
  m <- floor(nrow(res$bs)/thin)
  R <- matrix(0,nrow(res$Xf),m)
  for (i in 1:m) {
    f <- exp(res$Xf %*% res$bs[1+(i-1)*thin,1:ncol(res$Xf)])
    R[,i] <- getR(f,gamma,delta)$R
  }
  R
} ## sampleR

sensitivityR <- function(f,g=1:5,d=2:10,gamma=3,delta=5,
                         ylim=NULL,last.day=NULL) {
## Sensitivity plots w.r.t. parameters of SEIR model used for R
  R0 <- getR(f,1/gamma,1/delta)$R
  R0[1:delta] <- NA; R <- R0
  day <- 1:length(R0)-21
  xlim <- if (is.null(last.day)) range(day) else c(min(day),last.day)
  if (is.null(ylim)) ylim <- range(Rq)
  c1 <- 1.3
  plot(day,log(R),type="l",lwd=3,ylim=ylim,xlim=xlim,cex.lab=c1)

  for (i in 1:length(d)) {
    R <- getR(f,1/gamma,1/d[i])$R
    R[1:(1*d[i])] <- NA
    lines(day,log(R),col="grey")
  }

  for (i in 1:length(g)) {
    R <- getR(f,1/g[i],1/delta)$R
    R[1:delta] <- NA
    lines(day,log(R),col=4)
  }
  lines(day,log(R0),lwd=2);abline(v=11,col=2);abline(0,0,col=4)
} ## sensitivityR

plotR <- function(res,last.day=NULL,ylim=NULL,ylab="log(R)",times=c(-9,0,3,7)) {
## plot CI for R computed using infection profile and SEIR model
  R <- sampleR(res)
  n <- nrow(R); day <- 1:n-21
  Rq <- log(apply(R,1,quantile,prob=c(.025,.16,.5,.84,.975)))
  Rq[,1:5] <- NA
  xlim <- if (is.null(last.day)) range(day) else c(min(day),last.day)
  if (is.null(ylim)) ylim <- range(Rq)
  c1 <- 1.3
  plot(day,Rq[3,],type="l",ylim=ylim,xlim=xlim,ylab=ylab,cex.lab=c1)
  polygon(c(day,day[n:1]),c(Rq[1,],Rq[5,n:1]),col="lightgrey",border=NA)
  polygon(c(day,day[n:1]),c(Rq[2,],Rq[4,n:1]),col="grey",border=NA)
  lines(day,Rq[3,]);abline(v=11,col=2);
  if (!is.null(times)) for (i in 1:length(times)) abline(v=times[i],lty=i+1)
  abline(0,0,col=4)
} ## plotR

ps <- FALSE
if (ps) postscript("RCI.eps",height=4)
par(mfrow=c(1,2),mar=c(5,5,1,1))
plotR(res,last.day=70,ylim=c(-2,2))
sensitivityR(res$f[3,],last.day=70,ylim=c(-2,2))
if (ps) dev.off()
```