Supplementary files for:

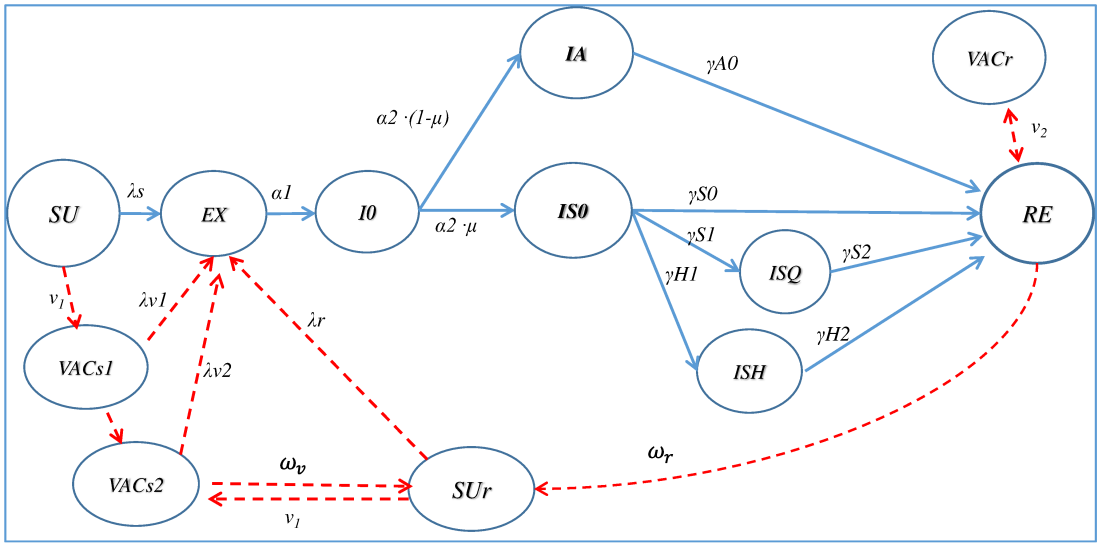
Vaccination against COVID-19 and society's return to normality in England: a modelling investigation of different types of naturally acquired and vaccine induced immunity

Song F, Bachmann MO. *BMJ Open* 2021; 11:e053507. doi: 10.1136/bmjopen-2021-053507

1. Model structure and status

This is a discrete-time population dynamic simulation model. Population in England are partitioned into discrete categories by sex (male or female), age (0-4, 5-9, then by 10 year age bands, and 80+), and Covid-19 infection categories. The main infection categories include: susceptible (SU), exposed (EX), infected (IN), and recovered (RE) (appendix figure 1). The infected individuals are further categorised as asymptomatic, symptomatic, self-isolated, and hospitalised.

Appendix figure 1: Model structure and transmission across states



Definitions of compartmental states in appendix figure 1

- SU: susceptible individuals
- SUr: Individuals susceptible to reinfection after immunity waning
- EX: exposed individuals, not infectious
- I0: infectious, before symptom onset
- IA: infectious individuals with no or very mild symptoms
- IS0: symptomatic patients who are not quarantined
- ISQ: symptomatic patients self-isolated
- ISH: symptomatic patients who are hospitalised
- RE: recovered from covid-19 infection
- VACs1: vaccinated (dose-1) susceptible individuals
- VACs2: vaccinated (dose-1) susceptible individuals
- VACr: vaccinated individuals who recovered from infection

Transition parameters in appendix figure 1:

- λ_s: Force of infection (λ) measures the risk (probability) of infection transmission.

Song F, Bachmann MO. *BMJ Open* 2021; 11:e053507. doi: 10.1136/bmjopen-2021-053507

- λ_{v1}: Force of infection (λ) after the first dose vaccine.
- λ_{v2}: Force of infection (λ) after the second dose vaccine.
- λ_r: Force of infection (λ) after waning of vaccine-induced or naturally acquired immunity.
- $\alpha 1$: rate of progressing from being exposed to being infectious.
- $\alpha 2$: rate of progressing from being asymptomatic infectious to symptomatic.
- μ: proportion of infected individuals who will be symptomatic; age-specific
- $\gamma A0$: rate of recovering for asymptomatic individuals
- $\gamma S0$ rate of recovering for symptomatic, not isolated/hospitalised patients
- $\gamma S1$: rate of being isolated in symptomatic patients
- $\gamma S2$: rate of recovering in isolated patients
- $\gamma H1$: rate of being hospitalised for symptomatic patients
- $\gamma H2$: rate of recovering in hospitalised patients
- ν_1: rate of vaccinating susceptible individuals
- ν_2: rate of vaccinating recovered individuals
- ω_v: rate of immunity waning in vaccinated individuals
- ω_r: rate of immunity waning in recovered individuals

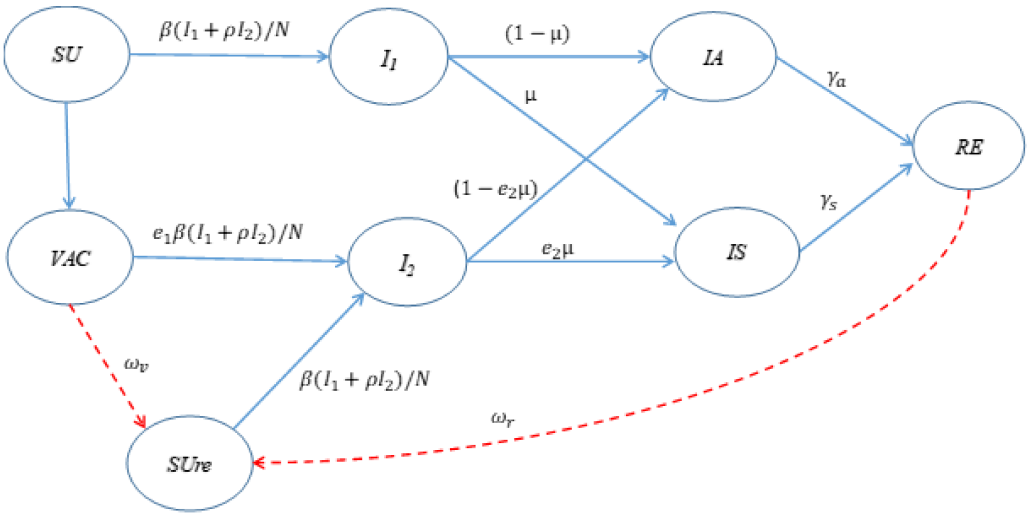
All individuals in England are assumed to be susceptible to SARS-CoV-2 infection at the beginning of 2020. Susceptible individuals may be infected by contacting infectious individuals, and the infection status is changed from "susceptible" (SU) to "exposed" (EX). The exposed individuals are not infectious during the early incubation period, but start to be infectious before the onset of symptoms. Individuals infected with SARS-CoV-2 virus may have no or very mild symptoms (IA), and palpable symptoms (symptomatic or clinical infections). Asymptomatic individuals can spread SARS-CoV-2 virus before recovery, although the transmission risk may be lower than symptomatic patients. Symptomatic patients are further classified into three categories: symptomatic patients who are neither isolated nor hospitalised (IS0), those who self-isolate at home (ISQ), and those who are hospitalised (including those being admitted to intensive care units) (ISH). Symptomatic patients are infectious and can transmit the virus to susceptible people before being isolated, hospitalised or recovered. We assume that hospitalised patients (ISH) are well isolated and no longer able to spread the virus to the susceptible population, although infectious patients who are self-isolated at home (ISQ) may transmit virus to household contacts.

Individuals may recover from previous infection of SARS-CoV-2 (RE), and the susceptible and recovered individuals may be vaccinated with vaccines again SARS-CoV-2 virus (VAC1 and VAC2). Individuals recovered or effectively vaccinated may develop immune responses against infection with SARS-CoV-2. However, if the protective immunity is not long lasting, individuals who have recovered or vaccinated may become susceptible again after the waning of the immunity (SUr).

Song F, Bachmann MO. *BMJ Open* 2021; 11:e053507. doi: 10.1136/bmjopen-2021-053507

The immune response against COVID-19, either by naturally acquired from past SARSE-CoV-2 infection or vaccine-induced, may be long lasting or short-lived. Immune response may reduce susceptibility of individuals to infection (infection protection, or sterilising, immunity), reduce disease severity after being infected (disease reduction immunity), and reduce infectivity of those who are reinfected after recovery or being vaccinated (reinfectivity reduction immunity).[1] According to existing evidence on immunological characteristics for other human coronaviruses, immunity against reinfection (sterilising immunity) may be waning in several months, while disease and reinfectivity reduction responses are likely long lasting.[1] According to these basic concepts specified by Lavine et al,[1] we incorporate the three types of immune responses into the model, to explicitly evaluate their impacts on future transmission dynamics (appendix figure 2).

Appendix figure 2: Illustration of immune responses by natural infection or vaccination



Notes to appendix figure 2:

- Appendix figure2 is a simplified version of appendix figure 1, not showing isolation and hospitalisation for symptomatic patients.
- *N*: The number of the population
- β: The transmission rate, i.e., the average number of individuals infected daily by an infectious person. It is a function of the number of daily close contacts per person (c), and the transmission risk per contact between a susceptible and an infectious individual (η): i.e., $\beta=c\cdot\eta$.
- I_1: Infectious individuals with primary infection
- I_2: Infectious individuals with secondary infection (infected after being vaccinated or recovered)
- ρ: Relative infectivity of the secondary infection (I_2) compared with the primary infection (I_1). For example, if ρ=0.6, the infectivity of I_2 is 40% lower than the infectivity of I_1
- μ: proportion of infected individuals who will be symptomatic; age-specific
- e_1: Relative efficacy of vaccine for sterilising immunity, reducing risk of virus transmission

- e_2: Relative efficacy of vaccine for pathology reduction, reducing the proportion of symptomatic cases after being infected
- *IA*: Asymptomatic individuals
- *IS*: Symptomatic patients
- γ_a : Average rate of recovering of asymptomatic individuals
- γ_s : Average rate of recovering of asymptomatic individuals
- ω_v: rate of immunity waning in vaccinated individuals
- ω_r: rate of immunity waning in recovered individuals

Overall and partial vaccine efficacy

Results of randomised controlled trials shown that vaccines may be >90% efficacious (e.g., Pfizer mRNA vaccine) in reducing severe symptomatic diseases, compared with the placebo group.[2] In appendix figure 2, e_1 and e_2 are parameters of vaccine's efficacy in blocking virus transmission and reducing symptomatic cases in the infected, respectively. The reduction in symptomatic cases in the vaccine group may be due to the prevention of infection in susceptible individuals (related to e_1), or a lower proportion of infected individuals being symptomatic in the vaccine group (related to e_2), or due to a combination of both (note: e_1 and $e_2 \geq 1$ - overall VE, and ≤ 1). Let λ is the transmission risk and μ is the proportion of symptomatic cases in the infected without vaccination. After being vaccinated, the transmission risk is reduced to $\lambda \cdot e_1$, and the proportion of symptomatic cases reduced to $\mu \cdot e_2$. For a vaccine with 90% efficacy in reducing the number of symptomatic cases (compared with the control group), it should be true that $e_1 \cdot e_2 = (1-0.90)$, or $e_1 = 0.10 / e_2$. There are many different possible combinations of e_1 and e_2 for a given overall efficacy in reducing symptomatic cases. For example, $e_1 = e_2 = SQRT(0.30) = 0.548$ corresponds to a 70% efficacy of vaccine with equal sterilising immunity and pathology reduction. If e_1=1 (i.e., zero efficacy in sterilising immunity), all vaccine efficacy will be attributable to the pathology reduction, with e_2=0.30, for a vaccine with 70% efficacy. The partial efficacy is calculated using: $Ei = 1 - (1 - Eo)/Ed$, where Ei is the partial efficacy for infection protection, Ed is the partial efficacy for disease reduction, and Eo is the overall vaccine efficacy. The equal partial efficacy is calculated by: $Ei = Ed = 1 - \sqrt{(1 - Eo)}$. For vaccines with 90%, 70% and 50% overall efficacy, the equal partial efficacy for the infection protection and disease reduction is 69.4%, 45.2%, and 29.3%, respectively.

Song F, Bachmann MO. *BMJ Open* 2021; 11:e053507. doi: 10.1136/bmjopen-2021-053507

2. Parameterisation, data sources, and simulation scenarios

2.1 Transition parameters and distribution of infectious period

In appendix figure 1, force of infection (λ) measures the risk of infection,[3] which is a function of transmission rate (η) and the prevalence of existing infectious individuals (*I*) among the population (N): $\lambda=\eta \cdot I/N$. The transmission rate η in the discrete-time model can be defined as the average number of new infected individuals generated daily by an infected person. That is, $\eta=Rt/T$, in which *Rt* is effective reproduction number and T is the average infectious period for infected individuals. We calculated η as a function of the number of daily contacts per person (c), and the risk of transmission per contact between a susceptible and an infected individual (β): η=c*β.[4]

The transition rate between model's compartments in infectious models is often assumed to be constant, calculated by 1/x, in which "x" is the average period that subjects remain before the transition to the model's next compartments.[3] Therefore, the infectious period in standard SIR or SEIR models is usually assumed to be exponentially distributed, with some limitations of the use of exponentially distributed infectious period.[5 6] In this study, we assumed that the transition probability between model's compartments are based on gamma distributed periods that individuals remain in a compartment.[7] The transition probability (y) at t is: $y_t = (cg_t - cg_{t-1})/(1 - cg_t)$, where cg_t is the gamma cumulative probability by the end of t. Given mean and shape (k) parameters, the gamma distribution based transition probability is used as a deterministic value to estimate the number of individuals moving between two status in this study.

2.2 Parameterisation and data sources

We estimated initial parameters based on relevant literature and data from the UK government websites (appendix table 1). Key parameters were calibrated according to the reported numbers of covid-19 deaths, hospitalised patients, and the prevalence of infected individuals in England from January 2020 to June 2021.[8]

Song F, Bachmann MO. *BMJ Open* 2021; 11:e053507. doi: 10.1136/bmjopen-2021-053507

Appendix table 1: Summary of key model parameters and data sources

Parameter	Value			Source and notes
Proportion of asymptomatic cases in infected individuals (%)				
Age: 0-9	0.71			Davis et al. [9]
10-19	0.79			
20-29	0.73			
30-39	0.67			
40-49	0.60			
50-59	0.51			
60-69	0.37			
70+	0.31			
Proportion of self-isolated symptomatic cases who are not hospitalised				
Age:	**0-49**	**50-69**	**70+**	Assumed by authors, and calibrated according to reported COVID-19 deaths.
Before 13/03/2021	0.10	0.10	0.10	
13/03-23/03/2020	0.40	0.60	0.80	
After 24/03/2020	0.80-0.90	0.80-0.95	0.90-0.95	
Estimated values of gamma distributed parameters (mean, shape k)				
	Before 12/03/20	**13/03-23/03/20**	**After 23/03/20**	Zhang et al. [10] Davies et al. [7] Ferguson et al. [11]
Duration from exposed to being preclinical infectious (day)	4.0 (4)	4.0 (4)	4.0 (4)	
Duration of preclinical infectious (day)	1.5 (2)	1.5 (2)	1.5 (2)	
Infectious period before recovery (day)	5.0 (5)	5.0 (5)	5.0 (5)	
Infectious period before being isolated (day)	4.0 (4)	3.0 (3)	2.0 (2)	
Infectious period during isolation at home (day)	2.0 (2)	3.0 (3)	4.0 (4)	
Infectious period before being hospitalised (day)	4.0 (4)	3.0 (3)	2.0 (2)	
Duration of hospitalisation (day)	10.0 (10)	10.0 (10)	10.0 (10)	
Delay from being infected to deaths from Covid-19 (day)	23.0 (23)	23.0 (23)	23.0 (23)	Verity et al. [12]
Case fatality rates (%)				
Age: 0-9	0.0026			Verity et al. [12]; Chinese CDC [13]
10-19	0.0148			
20-29	0.0600			
30-39	0.1460			
40-49	0.2950			
50-59	1.2500			
60-69	3.9900			
70-79	8.6100			
80+	13.400			
Age specific rates of hospitalisation among infected individuals				
0-9	0.010%			Verity et al. [12]
10-19	0.041%			
20-29	1.04%			
30-39	3.43%			
40-49	4.25%			
50-59	8.16%			
60-69	11.8%			

Song F, Bachmann MO. *BMJ Open* 2021; 11:e053507. doi: 10.1136/bmjopen-2021-053507

70-79	16.6%	
80+	18.4%	
SARS-CoV-2 transmission related parameters		
Initial importing of infected cases	The first exposed case was imported to England on 15 January, and the daily number of infectious cases imported was increased by one until 9 February 2020, with a total number of 351 cases imported in 25 days.	Assumed by authors and calibrated with the observed number of covid-19 deaths in England.
the risk of transmission per contact between a susceptible and an infected individual (β)	Before any NPI measures: 0.094 NPI measures (03/2020): 0.066 Alpha variant (10/2020-): 0.077 Delta variant (06/2021-): 0.081	Calibrated according to changes in reported R values,[14] NPI policies[15] and reported covid-19 deaths.[8]
Seasonal changes in transmission risk (09/2021-12/2024)	10% higher in September, October, March and April, and 20% higher in November, December, January and February, compared with the risk in summer months from May to August.	Literature (e.g.,[16]) and authors' assumptions.
Transmission risk by asymptomatic individuals	21% of the transmission risk by symptomatic cases	Li et al.[17]

Assumed impacts of NPIs on general contacts in England. Notes:
Notes: Values are scaling fractions to reduce the normal contacts. For example, a fraction of 0.80 means the contacts are reduced by 20%.

Age group	**0-19**	**20-59**	**60-69**	**70+**	
Before 13/3/2020	1.00	1.00	1.00	1.00	The sex-and-age-specific numbers of daily contacts per person from Mossong et al.[18] Calibrated according to covid-19 control restrictions[15] and reported covid-19 deaths in England.[8]
13/03/2020-	0.90	0.80	0.70	0.60	
17/03/2020-	0.80	0.70	0.60	0.50	
24/03/2020-	0.40	0.30	0.20	0.15	
05/07/2020-	0.60	0.55	0.40	0.20	
01/09/2020-	0.80	0.60	0.50	0.30	
05/11/2020-	0.60	0.30	0.20	0.15	
02/12/2020-	0.70	0.60	0.50	0.30	
05/01/2021-	0.40	0.30	0.20	0.15	
08/03/2021-	0.80	0.60	0.50	0.30	
01/05/2021	0.90	0.80	0.70	0.60	
19/07/2021-	1.00	1.00	1.00	1.00	

Vaccine relative efficacy in reducing symptomatic cases
Notes: We assume that vaccine efficacy for reducing symptomatic cases is equally attributable to infection reduction and disease reduction in the main projections.

14 days after the first dose	62.5%	Public Health England [19]
After the second dose	85.0%	
Long-term efficacy after revaccination	62.5%, 85.0%	Public Health England,[19] and authors' assumption.
Durability of naturally acquired sterilising immunity	365 or 730 days	Lavine et al,[1] Le Bert et al,[20] Widge et al.[21]
Durability of vaccine-induced sterilising immunity	182 or 365 days	
Durability of immunity against severe disease after re-infection	>4 years	
Reduction in reinfectivity (transmission risk) in vaccinated or recovered individuals	30%, 45% or 60%	Mallapaty et al,[22] PHE.[19]

Vaccination programme related parameters Notes: The mass vaccination is modelled as an age-based phase approach, starting from people aged ⩾70, followed by individuals aged 60-69, 50-59, 30-49, and then those aged 16-29 years old.		
Maximum number of vaccinated individuals per day in England	180,000	According to data on numbers of vaccinated individuals[8]
Interval between the first and second vaccine dose	9 weeks	PHE[19]
Vaccination coverage	75%, 80%, 85% and 90%, respectively, in adults aged 16-29, 30-39, 40-49, and ⩾50 years old.	Reported vaccination data[8 19] and authors' assumption.
Vaccination coverage sensitivity analyses	Lower (60%, 70%, and 80%) and higher (80%, 85%, and 90%) in people aged 16-29, 30-39, and 40-49 years old, respectively.	Authors' assumptions.
Frequency of revaccination programmes	Number of (days between) repeated vaccination programmes during 2021-2024: 1 (…), 2 (365), 3 (365); 2 (730), 3 (487), 4 (365), 5 (292), 6 (243)	Authors' assumption

We obtained population statistics in England (estimates of mid-year 2020) from Office for National Statistics. It was assumed that all individuals in England were susceptible to SARS-CoV-2 infection at the beginning of 2020. By contacting with infectious individuals, susceptible individuals may be infected, and their infection category is changed from "susceptible" (SU) to "exposed" (EX). "Exposed" refers to the pre-infectious status of infected individuals. According to data from previous studies, the period of incubation before symptom onset was on average 5.5 days,[10] and the exposed individuals start to be infectious about 1.5 days before the onset of symptoms.[7 11] Therefore, we assumed a gamma distribution of incubation period, with a mean non-infectious period of 4 days (k=4.0) after being exposed, and a mean infectious period of 1.5 days (k=2) before symptom onset (appendix table 1).

Individuals infected with SARS-CoV-2 virus may have no or very mild symptoms (asymptomatic infected), and palpable symptoms (symptomatic patients). As in previous modelling studies [7 11], it was assumed that asymptomatic individuals can spread SARS-CoV-2 virus before recovery, although the infectious risk was assumed to be 21% of symptomatic patients.[17] We used age-specific rates of asymptomatic cases in the infected individuals, reported in a study based on data from 6 countries (appendix table 1).[9]

Symptomatic patients are further classified into three categories: symptomatic patients who are neither isolated nor hospitalised (mainly at the initial phase of the epidemic), those who are self-isolated at home, those who are hospitalised (see appendix figure 1). We assume that asymptomatic individuals were not isolated, although the average number of daily contacts could be reduced by non-pharmaceutical interventions (NPIs), including social distancing, testing, contact tracing, and lockdown. Assumed proportions of self-isolation of symptomatic cases who are not hospitalised, depending on age and NPI measures are shown in appendix table 1.

Song F, Bachmann MO. *BMJ Open* 2021; 11:e053507. doi: 10.1136/bmjopen-2021-053507

We assume that only symptomatic patients are hospitalised, and age specific rates of hospitalisation among symptomatic individuals were from Verity et al.[12] The hospitalisation rates were calibrated according to reported numbers of hospitalised patients with covid-19 in England.[23] Based on the reported number of hospitalised patients and estimated number of symptomatic cases, the hospitalisation rate was estimated to be 70% lower than the estimated by Verity et al.[12] Symptomatic patients are infectious and can transmit the virus to susceptible people before being hospitalised or isolated. We assume that hospitalised patients are no longer able to spread the virus to susceptible individuals in the community. However, infected individuals who are self-isolated at home may transmit virus to household contacts. The infectious period before recovery was assumed to be gamma distributed, with a mean value of 5 days. Before implementing any NPIs, the infectious period of symptomatic cases was of a mean value of 4 days (k=4) before being quarantined or hospitalised. After implementing NPI measures, the infectious period for isolated and hospitalised patients was reduced, having a mean value of 2 days (k=2). The mean hospital stay was assumed to be 10 days (k=10) (including ICU admitted patients) (appendix table 1). Verity et al estimated that the average duration from symptom onset to death was 17.8 days.[12] Therefore, we assume that covid-19 related deaths occur on average 23 days (k=23) after being exposed/infected.

The simulation starts from 1 January 2020, over a period of five years until the end of 2024. We assume that the first exposed case was imported to England on January 15th 2020, and the daily number of infectious cases imported was increased by one until 9 February 2020, with a total number of 351 cases imported in 25 days. The sex-and-age-specific numbers of household and community daily contacts per person in the UK were obtained from a study in 8 European countries.[18 24] For the purpose of simplicity, we considered only household contacts (relevant to self-isolation at home) and general daily contacts (for all types of contacts). The risk of positive transmission per contact between susceptible and infectious individuals (β) was estimated by calibrating estimated and reported numbers of covid-19 deaths in England, household and general daily contacts per person, and other model parameters.

In this study, all COVID-19 related deaths are assumed to be from symptomatic cases, and age specific case fatality rates were based on a study by Verity et al.[12] We assume that individuals infected with Covid-19 will not die from other causes before recovery. Average sex and age specific rates of all-cause deaths in England during 2015-2019 [25] were applied to people who are not infected with or recovered from covid-19. For simplicity and maintaining a stable population, we assumed that the number of births at day t equals to the number of all deaths at day t-1. Furthermore, we did not consider the influence of migration on the population. We adjusted the number of individuals belong to an age group (all <80+) at the beginning of the year since 2021 by shifting 20% (for age group 0-4 and 5-9) or 10% (for age group 10-19, … 70-79) of them to the adjacent higher age group.

Song F, Bachmann MO. *BMJ Open* 2021; 11:e053507. doi: 10.1136/bmjopen-2021-053507

NPI and seasonal impacts on transmission parameters

Since March 2020, NPI measures were recommended and gradually tightened in England, including hand washing, mouth covering when coughing in public places, home isolation of individuals with COVID-19 like symptoms, shielding of vulnerable individuals, avoiding non-essential contacts, and maintaining social distancing. These measures reduced contacts and transmission risk, and shortened the period of transmission by symptomatic individuals. We assumed that the general population's contact rates were reduced by 10% to 40%, depending on age and co-morbidity. Based on the reported number of COVID-19 deaths, we estimate that the transmission risk per contact between infectious and susceptible individuals was reduced from β=0.094 before the implementation of any NPIs to 0.069 by 15 March 2020 and 0.062 since May 2020. The UK government put lockdown measures in place from 24 March 2020, including working from home if possible, closure of schools and non-essential shops, pubs and restaurants, avoiding non-essential travelling, and cancelling gathering activities. We assume that numbers of general population contacts were reduced by 60-85% (appendix table 1). We assume that the household contacts were not influenced by the NPI measures.

The lockdown measures in England started to be relaxed from 13 May 2020 by allowing partial returning to work. Further relaxing of control measures followed, including reopening of some shops and allowing outdoor meetings up to six people from 1 June, re-opening of more non-essential shops from 5 June, and further relaxing of restrictions (such as re-opening of pubs and restaurants) from 5 July 2020. However, social distancing measures was maintained and face covering was required where social distancing could not be implemented. From 1 September 2020, schools in England were re-opening. Consequently, the transmission risk per contact between susceptible and infectious individuals was increased since September. The impacts of these changes in NPIs were reflected in the assumed social contacts and transmission risk. Because of the new virus variants,[26] the average transmission risk per contact was increased β=0.077 since October 2020, and β=0.081 since June 2021.

To incorporate the impact of seasonality on future projections, we assumed that the transmission risk is increased by 10% in September, October, March and April, and increased by 20% in November, December, January and February.

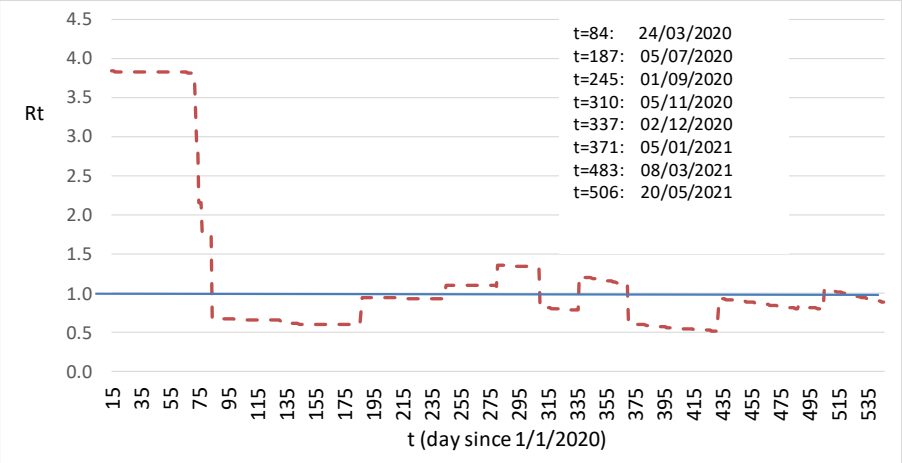
Model verification/calibration

We used the developed model and initially estimated parameters to simulate the covid-19 epidemic in England from January 2020 to June 2021. Key parameters were calibrated based mainly on reported covid-19 related deaths, although numbers of hospitalised patients and infection rates in England were also considered.

We assume that the first exposed case was imported to England on 15 January 2020, and the number of cases imported each day increased by one more case than the previous day until 9 February 2020 (the total number of cases imported in 25 days was therefore 351). We don't use the reproduction number

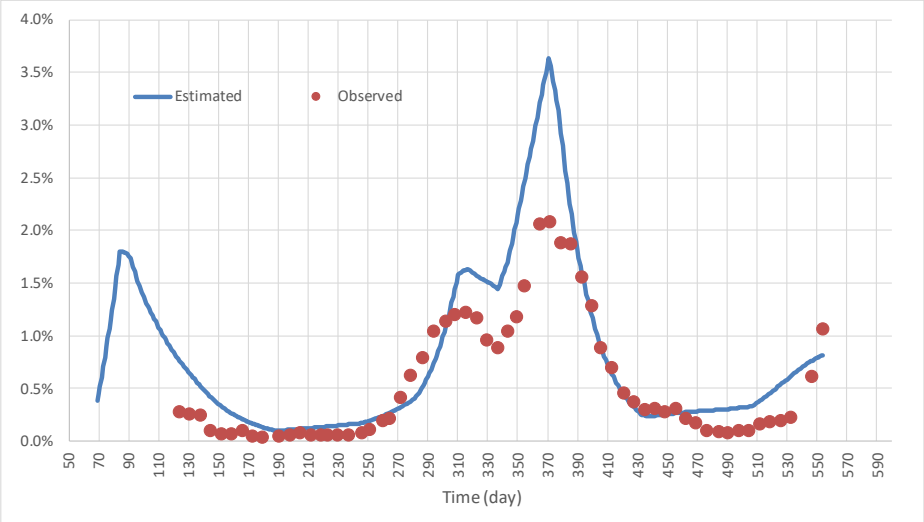
(R0 or Rt) as an input parameter, but derived the reproduction numbers based on a method used by Giordano and colleagues (see equation 55 in Mathematical equations) [3 27]. We estimated that the basic reproduction number (R0) was 3.63 at the initial stage of the COVID-19 epidemic before any control measures were taken in England, which is similar to findings from previous studies.[7 11] Following the implementation of NPI measures, the estimated reproduction value (Rt) was reduced to 0.65 by 24 March. The Rt value was increased to 0.93 by 5 July 2020 after the NPI measures were relaxed, and Rt was about 1.09 after school reopening in September and 1.31 by October 2020. The Rt was reduced to 0.79 since 5 November 2020 after reintroducing NPI measures, increased to about 1.17 after relaxing NPIs since 2 December 2020, and reduced again to about 0.57 since 5 January 2021 after reintroducing lockdown measures (plus rolling out of vaccination) (appendix figure 3). The estimated R values were within the range of the reported in England (https://coronavirus.data.gov.uk/).

Appendix figure 3: Estimated reproduction numbers, between 01/2020 – 06/2021, in England



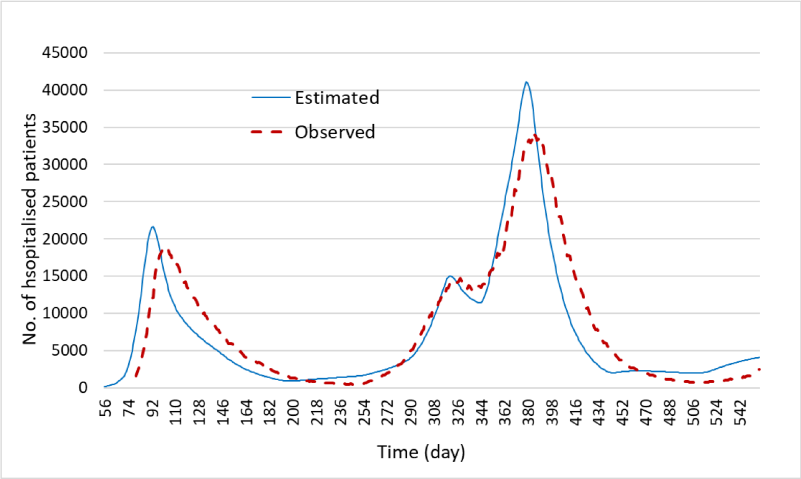
The model estimated that the prevalence of the recovered was 5.5 by 26 April, 7.6% by 24 May, and 8.5% by 24 June 2020, which were similar to the estimated rates of positive antibodies to Covid-19 in the UK (i.e., 7.1% in May-June 2020).[28] Data on the prevalence of infected individuals in the community was available from May 2020. The model estimated prevalence of infected individuals from January 2020 to January 2021, which had a similar trend as the reported prevalence in England (appendix figure 4).

Appendix figure 4: Estimated and reported prevalence of infection, from January 2020 to June 2021, in England



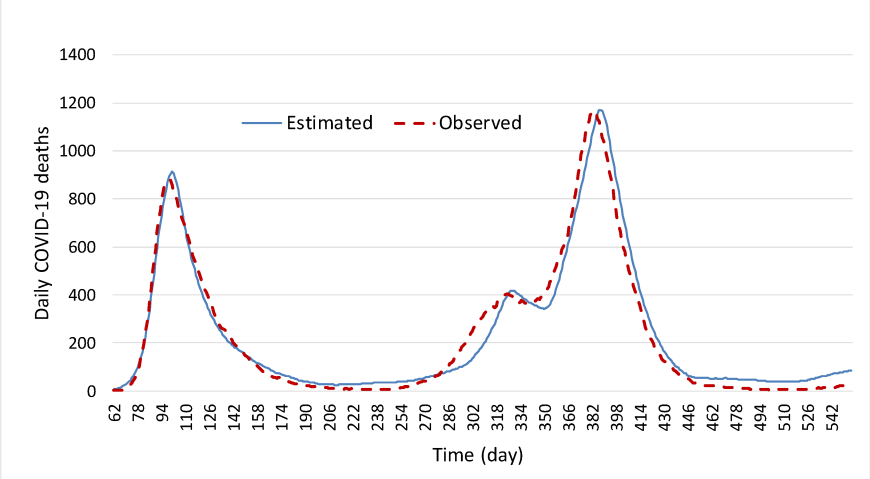
Changes in the estimated numbers of hospitalised COVID-19 patients were of similar trends as the reported numbers of hospitalised patients during 01/2020-01/2021. However, there were considerable differences at peak time points (appendix figure 5), which may be explained by reduced hospitalisation rates during peak period.

Appendix figure 5: Reported and estimated numbers of hospitalised Covid-19 patients, during 01/2020-01/2021, in England



Appendix figure 6 shows that the estimated daily deaths well matched the observed daily deaths from Covid-19, from January 2020 to June 2021, in England.

Appendix figure 6: Reported and estimated daily deaths from Covid-19, during 01/2020-06/2021, in England



3. Vaccination and projection scenarios

We used estimates of transmission parameters, age-specific hospitalisation rates and case fatality rates in June 2021 to project COVID-19 deaths from 2021 to 2024, under various scenarios of vaccine efficacy, durability of both naturally acquired and vaccine induced immunity, and reduction in reinfectivity. There are no more restrictions on social activities since 19 July 2021 in England, and social contacts are return to normal as before the pandemic, but basic hygienic measures would be maintained.

The UK Government's Vaccination Taskforce has recommended rollout of vaccines against COVID-19 to be prioritised primarily by age and comorbidity, with older people being vaccinated first.[29] The Joint Committee on Vaccination and Immunisation (JCVI) in the UK previously recommended COVID-19 vaccination of individuals aged ≥18, and also recommended vaccination of young people aged 16-17 years old on 4 August 2021.[30] Vaccination of prioritised individuals began from 8 December 2020 in the UK and around 2 million individuals were vaccinated (mostly with a single dose of Pfizer vaccine) by 10 January 2021.[31] By July 2021, the coverage of individuals who were fully vaccinated (with two does) was >90% in adults aged ≥65, 80-90% in those aged 50-64, 60% in adults aged 40-49.[32] For young adults aged 18-39, about 36% have received the first dose of vaccines by July 2021. Therefore, the vaccination coverage in England has been high in older adults, but it remains uncertain whether such high coverage could be achieved in younger adults. In this study, the mass vaccination is modelled as an age-based phase approach, starting from people aged ≥70, followed by individuals aged 60-69, 50-59, 30-49, and then those aged 16-29. We assume that the mass vaccination starts from 1 January 2021, and the maximum number of individuals vaccinated per day is 180,000 in England, to matched with numbers of vaccinated individuals according to the official statistics. In the main analysis, we assume that the uptake rate is 75%, 80%, 85% and 90%, respectively, in adults aged 16-29, 30-39, 40-49, and ≥50 years old. Because of uncertain coverage of vaccination in younger people, we conducted sensitivity analyses under scenarios with lower (60%, 70%, and 80%) and higher (80%, 85%, and 90%) coverage of vaccination in people aged 16-29, 30-39, and 40-49 years old, respectively.

Results of randomised controlled trials shown that vaccines may be >90% efficacious (e.g., Pfizer mRNA vaccine) in reducing severe symptomatic diseases, compared with the placebo group. Assume that e_1 and e_2 are parameters of vaccine's efficacy in blocking virus transmission and reducing symptomatic cases in the infected, respectively (as in appendix figure 2). The reduction in symptomatic cases in the vaccine group may be due to the prevention of infection in susceptible individuals (i.e., infection protection, related to e_1), or a lower proportion of infected individuals being symptomatic in the vaccine group (i.e., disease reduction, related to e_2), or due to a combination of both. For a vaccine with 90% efficacy in reducing the number of symptomatic cases (compared with the control group), it is true that $e_1 \cdot e_2 = (1-0.90)$, or $e_1 = (1-0.90)/e_2$ (note: e_1 and e_2 ≥1-overall VE, ≤1). There are many different possible combinations of e_1 and e_2 for an overall efficacy in reducing symptomatic cases. We assume that vaccine efficacy is equally attributable to infection and disease reduction in the main

Song F, Bachmann MO. *BMJ Open* 2021; 11:e053507. doi: 10.1136/bmjopen-2021-053507

projections. The infection protection (sterilising immunity) after vaccination has been demonstrated. For example, an observational study in the UK (SIREN) found that the risk of being infected was reduced by 70% in health workers after a single dose of the Pfizer-BioNTech vaccine.[33 34]

Both Pfizer-BioNTec and AstraZeneca vaccines are 2-dose regimens, the policy in the UK has been to initially provide the first dose to as many individuals as possible to maximise the public health impact.[35] Exploratory assessment of data from clinical trials found that the short-term vaccine efficacy from the first dose of the Pfizer-BioNTech vaccine and the AstraZeneca vaccine is about 90% and 70%, respectively.[35] Public Health England (PHE) in July 2021 estimated that the efficacy was 55-70% after the first dose, and 70-85% or 85-95% after the second dose.[32] Therefore, we assume that the overall vaccine efficacy is 62.5% after the first dose and 85% after the second dose. The protection effects start 14 days after the first dose vaccination, and the interval between the first and second dose is 9 weeks. For vaccines with 62.5% and 85.0% overall efficacy, the equal partial efficacy for the infection protection and for disease reduction is 38.8% and 61.3%, respectively. The overall vaccine efficacy after the second dose may be lower than 85.0% due to new variants of SARS-CoV-2 virus. Therefore, we evaluated two overall vaccine efficacy rates, 62.5% or 85.0%, after the second dose of vaccine.

Available evidence has indicated that the duration of sterilising (infection protection) immunity after coronavirus infection ranges from 0.5 to two years.[1] Serum neutralizing antibodies were detected in all participants at four months follow up after SAR-CoV-2 mRNA vaccination.[21] Therefore, we assume that naturally acquired sterilising immunity lasts for 365 or 730 days, and vaccine-induced sterilising immunity lasts for 182 or 365 days. After waning of sterilising immunity, individuals may be susceptible again to infection with SARS-CoV-2 virus, but the disease reduction immunity is likely longer lasting.[1] Due to the existence of disease reduction immunity, the reinfectivity of individuals who are reinfected after waning of sterilising immunity may be reduced. Lavine and colleagues estimated that the secondary transmissibility (i.e., reinfectivity) was 35% of the primary transmissibility (i.e., the reinfectivity was reduced by 65%).[1] Evidence from clinical trials and vaccination in the real world indicated that the viral loads and the duration of virus shedding in the infected individuals after vaccination were considerably reduced, compared with unvaccinated individuals.[36 37] Based on preliminary data, PHE estimated that the reinfectivity was reduced by 35-50% after the first dose of vaccines.[32] The reduction in reinfectivity is likely to be larger after the second dose of vaccines. More recent studies reported that fully vaccinated individuals who were infected were up to 78% less likely to transmit the virus to unvaccinated individuals.[22] Therefore, we assume a range of the risk of reinfectivity after waning of sterilising immunity; the reinfectivity is reduced by 30%, 45% or 60%. We also assume that the infectivity of ineffectively vaccinated individuals is the same as recovered individuals whose sterilising immunity has waned, and that vaccination of individuals recovered from natural infection boosts their naturally acquired immunity.

Song F, Bachmann MO. *BMJ Open* 2021; 11:e053507. doi: 10.1136/bmjopen-2021-053507

We considered different frequent scenarios of revaccination programmes. First, we evaluated a single vaccination programme and multiple (2-4) annual vaccination programmes during 2021-2024. In addition, simulation projections were conducted by revaccination programmes with different intervals, including 2-6 revaccination programme and corresponding intervals between them.

In summary, projection scenarios are defined from the following aspects: vaccine efficacy, frequency of revaccination programmes, durability of natural and vaccine induced sterilising immunity, reduction in reinfectivity after the waning of natural and vaccine-induced immunity against reinfection. The main characteristics of the simulated scenarios are available in supplementary table 1, supplementary table 2 and supplementary table 3.

In this study, we focus on deaths in people infected with COVID-19, although our model also produces estimates of changes in effective reproduction values (Rt), numbers of infected and vaccinated individuals, and hospitalised patients. We performed multiple simulations under various scenarios. For clarity, we focus on results of selected scenarios in the main text, but report more data on simulation results in supplementary tables.

Song F, Bachmann MO. *BMJ Open* 2021; 11:e053507. doi: 10.1136/bmjopen-2021-053507

4. Model's mathematical equations

Notations:

- subscript used: "s" refers to sex, 1: male, 2: female, 3: both male and female; "a" refers to age group, 1: 0-4 years, 2: 5-9 years, 3: 10-19, …, 10: ≥ 80; 11: all age groups. "t" refers to time (day).
- N: The number of the population
- λ_s: Force of infection (λ) measures the risk (probability) of infection transmission, which is a function of transmission rate (β) and the prevalence of infectious individuals (I) among the population (N): $\lambda=\beta \cdot I/N$. [3]
- β: The transmission rate β in this discrete-time model is defined as the average number of individuals infected daily by an infectious person. It is a function of the number of daily close contacts per person (c), and the transmission risk per contact between a susceptible and an infectious individual (η): i.e., $\beta=c \cdot \eta$. [4]
- $\alpha 1$: rate of progressing from being exposed to being infectious.
- $\alpha 2$: rate of progressing from being asymptomatic infectious to symptomatic.
- μ: proportion of infected individuals who will be symptomatic; age-specific
- *infA*: The fraction of infection force for infected individuals with no or mild symptoms. It was assumed that infA=0.5 in this study.
- fS0: fraction of symptomatic patients who will not be quarantined.
- fSq: fraction of symptomatic patients who will be quarantined (self-isolation).
- fSh: fraction of symptomatic patients who will be hospitalised (including ICU admission).
- $\gamma A0$: rate of recovering for asymptomatic individuals
- $\gamma S0$ rate of recovering for symptomatic, not isolated/hospitalised patients
- $\gamma S1$: rate of being isolated in symptomatic patients
- $\gamma S2$: rate of recovering in isolated patients
- $\gamma H1$: rate of being hospitalised for symptomatic patients

  $\gamma YH2$: rate of recovering in hospitalised patients
- ν_1: rate of vaccinating susceptible individuals
- ν_2: rate of vaccinating recovered individuals
- ρ: Relative infectivity of the secondary infection (I_2) compared with the primary infection (I_1). For example, if $\rho=0.6$, the infectivity of I_2 is 40% lower than the infectivity of I_1
- e_1: Relative efficacy of vaccine for sterilising immunity, reducing risk of virus transmission
- e_2: Relative efficacy of vaccine for pathology reduction, reducing the proportion of symptomatic cases after being infected
- IA: Asymptomatic individuals
- IS: Symptomatic patients
- γ_a : Average rate of recovering of asymptomatic individuals
- γ_s : Average rate of recovering of asymptomatic individuals

- ω_v: rate of immunity waning in vaccinated individuals
- ω_r: rate of immunity waning in recovered individuals
- $drOth_{s,a,t}$: sex, age-specific risk of deaths from causes other than covid-19, specific by week of the year.
- $drCov_{s,a,d}$: death risk from infected individuals before recovery, specific according to days since being infected.
- ds0, dsq, and dhos are the proportion of covid-19 deaths among symptomatic patients who are not quarantined, those who are isolated, or hospitalised, respectively. 1=ds0+dsq+dhos

Sex and age specific population:

$$\boldsymbol{N_{s,a,t}} = SU_{s,a,t} + SUr_{s,a,t} + VAC1_{s,a,t} + EX1_{s,a,t} + I01_{s,a,t} + EX2_{s,a,t} + I02_{s,a,t} + IA_{s,a,t} + IS0_{s,a,t} + ISQ_{s,a,t} + ISH_{s,a,t} + RE_{s,a,m,t} \quad (1)$$

Total number of the primary infection with no symptoms (age-specific):

$$aIA1_{a,t} = \sum_s (IA1_{s,a,t} + I01_{s,a,t} \cdot (1 - \mu_a)) \quad (2)$$

Total number of the primary infections with symptoms, isolated (age-specific):

$$aISQ1_{a,t} = \sum_s (ISQ1_{s,a,,t}) \quad (3)$$

Total number of the primary infections with symptoms, not isolated (age-specific):

$$aIS01_{a,t} = \sum_{s,} (IS01_{s,a,t} + I01_{s,a,t} \cdot \mu_a) \quad (4)$$

Total number of the secondary infection with no symptoms (age-specific):

$$aIA2_{a,t} = \sum_s (IA2_{s,a,t} + I02_{s,a,t} \cdot (1 - \mu_a \cdot e_2)) \quad (5)$$

Total number of the secondary infections with symptoms, isolated (age-specific):

$$aISQ2_{a,t} = \sum_s (ISQ2_{s,a,,t}) \quad (6)$$

Total number of the secondary infections with symptoms, not isolated (age-specific):

$$aIS02_{a,t} = \sum_{s,} (IS02_{s,a,t} + I02_{s,a,t} \cdot \mu_a \cdot e_2) \quad (7)$$

Sex and age specific susceptible population:

$$\boldsymbol{SU_{s,a,t+1}} = (SU_{s,a,t} - su\text{Exp}_{s,a,t} - VAC1_{s,a,t}) \cdot (1 - drOth_{s,a,t}) + NewBirth_{s,t} \quad (8)$$

Note: $drOth_{s,a,,t}$ is sex, age-specific death rates for non-covid causes, specific by week of the year.

Song F, Bachmann MO. *BMJ Open* 2021; 11:e053507. doi: 10.1136/bmjopen-2021-053507

Newly exposed/infected with SARS-CoV-2 in susceptible individuals:

$$suExp_{s,a,t} = \sum_{j=1}^{10} SU_{s,a,t} \cdot \eta_t \left(\left((Ca_{a,j,t} \cdot \frac{infA \cdot aIA1_{j,t} + aIS1_{j,t}}{N_{3,j,t}} + Cb_{a,j,t} \cdot \frac{aISQ1_{j,t}}{N_{3,j,t}}) \right) + \rho \cdot \left(Ca_{a,j,t} \cdot \frac{infA \cdot aIA2_{j,t} + aIS2_{j,t}}{N_{3,j,t}} + Cb_{a,j,t} \cdot \frac{aISQ2_{j,t}}{N_{3,j,t}} \right) \right) \quad (9)$$

Notes: $Ca_{a,j,t}$ is the average number of general contacts between people aged a and j; and $Cb_{a,j,t}$ is the average number of household contacts between people age a and j.

Newly exposed/infected in vaccinated individuals:

$$vacExp_{s,a,t} = \sum_{j=1}^{10} VAC1_{s,a,t} \cdot \eta_t \cdot e_1 \left(\left((Ca_{a,j,t} \cdot \frac{infA \cdot aIA1_{j,t} + aIS1_{j,t}}{N_{3,j,t}} + Cb_{a,j,t} \cdot \frac{aISQ1_{j,t}}{N_{3,j,t}}) \right) + \rho \cdot \left(Ca_{a,j,t} \cdot \frac{infA \cdot aIA2_{j,t} + aIS2_{j,t}}{N_{3,j,t}} + Cb_{a,j,t} \cdot \frac{aISQ2_{j,t}}{N_{3,j,t}} \right) \right) \quad (10)$$

Newly exposed/infected in the recovered or vaccinated after waning of immunity:

$$sureExp_{s,a,t} = \sum_{j=1}^{10} SUre_{s,a,t} \cdot \eta_t \left(\left((Ca_{a,j,t} \cdot \frac{infA \cdot aIA1_{j,t} + aIS1_{j,t}}{N_{3,j,t}} + Cb_{a,j,t} \cdot \frac{aISQ1_{j,t}}{N_{3,j,t}}) \right) + \rho \cdot \left(Ca_{a,j,t} \cdot \frac{infA \cdot aIA2_{j,t} + aIS2_{j,t}}{N_{3,j,t}} + Cb_{a,j,t} \cdot \frac{aISQ2_{j,t}}{N_{3,j,t}} \right) \right) \quad (11)$$

The number of the recovered or vaccinated who lose sterilising immunity (d from 1 to tt):

$$SUre_{s,a,t+1} = (SUre_{s,a,t} - sureEXP_{s,a,t}) \cdot (1 - drOth_{s,a,t}) + \sum_{d=1}^{tt} \left(iRE_{s,a,d,t} \cdot \omega_{r,d} + iVAC1_{s,a,d,t} \cdot \omega_{v,d} \right) \quad (12)$$

Notes: "*tt*" is the total number of days simulated. $\omega_{r,d}$ and $\omega_{v,d}$ are gamma distributed rate of immunity waning, respectively, a function of days since the recovery and vaccination. $iRE_{s,a,d,t}$ is the number of recovered since d days from recovery; and $eVAC1_{s,a,d,t}$ is the number of vaccinated since d days after vaccination.

The number of new (d=1) primary infections in susceptible individuals:

$$iEX1_{s,a,1,t} = suEXP_{s,a,t} \quad (13)$$

The number of new (d=1) secondary infections in recovered or vaccinated individuals:

$$iEX2_{s,a,1,t} = sureEXP_{s,a,t} + vacExp_{s,a,t} \quad (14)$$

For d=1,2,3…60 (assuming all will be dead or recovered by day 60 after being infected):

$$iEX1_{s,a,d+1,t+1} = iEX1_{s,a,d,t} \cdot (1 - \alpha 1_d) \quad (15)$$

$$iEX2_{s,a,d+1,t+1} = iEX2_{s,a,d,t} \cdot (1 - \alpha 1_d) \quad (16)$$

Song F, Bachmann MO. *BMJ Open* 2021; 11:e053507. doi: 10.1136/bmjopen-2021-053507

Assumed that covid-19 deaths were from symptomatic patients only. Overall deaths from covid-19 were calculated using the case fatality rates, and timing of covid-19 related deaths were assumed to have a gamma distribution according to days since being infected. Therefore a variable was introduced to record number of symptomatic individuals by days since being exposed/infected to calculate number of covid-19 deaths:

$$iSY_{s,a,1,t} = \mu_a\left(suExp_{s,a,t} + e_2 \cdot (surExp_{s,a,t} + vacExp_{s,a,,t})\right) \quad (17)$$

For d=1,2,3…60 (the transmission completed by day 60):

$$iSY_{s,a,d+1,t+1} = iSY_{s,a,d,t} \cdot (1 - drCov_{s,a,d}) \quad (18)$$

The number of covid-19 deaths at time t:

$$dthCov_{s,a,t} = \sum_{d=1}^{60} iSY_{s,a,d,t} \cdot drCov_{s,a,d} \quad (19)$$

The number of new (d=1) primary infections individuals before onset of symptoms:

$$iI01_{s,a,1,t+1} = \sum_{d=1}^{60} iEX1_{s,a,d,t} \cdot \alpha 1_d \quad (20)$$

The number of new (d=1) secondary infectious individuals before onset of symptoms:

$$iI02_{s,a,1,t+1} = \sum_{d=1}^{60} iEX2_{s,a,d,t} \cdot \alpha 1_d \quad (21)$$

For d=1,2,3…60 (the transmission completed by day 60):

$$iI01_{s,a,d+1,t+1} = iI01_{s,a,d,t} \cdot (1 - \alpha 2_d) \quad (22)$$

$$iI02_{s,a,d+1,t+1} = iI02_{s,a,d,t} \cdot (1 - \alpha 2_d) \quad (23)$$

The number of all infectious individuals before onset of symptoms:

$$\boldsymbol{I0_{s,a,t}} = \sum_{d=1}^{60} iI0_{s,a,d,t} \quad (24)$$

The number of new (d=1) infected individuals with no or very mild symptoms:

$$iIA_{s,a,1,t+1} = \sum_{d=1}^{60}\left(iI01_{s,a,d,t} \cdot \alpha 2_d \cdot (1 - \mu_a) + iI02_{s,a,d,t} \cdot \alpha 2_d \cdot (1 - e_2\mu_a)\right) \quad (25)$$

For d=1,2,3…60 (the transmission completed by day 60):

$$iIA_{s,a,d+1,t+1} = iIA_{s,a,d,t} \cdot (1 - \gamma A_d) \quad (26)$$

The number of all infectious individuals with no or mild symptoms:

$$IA\_(s,a,t) = \sum_{d=1}^{60} iIA_{s,a,d,t} \quad (27)$$

The number of all new (d=1) symptomatic patients:

$$SYM_{s,a,t+1} = \sum_{d=1}^{60}(iI01_{s,a,d,t} + e_2 \cdot iI02_{s,a,d,t}) \cdot \alpha 2_d \cdot \mu_a \quad (28)$$

The number of new (d=1) symptomatic patients who are not self-isolated:

$$iIS0_{s,a,1,t} = SYM_{s,a,t} \cdot fS0_t \quad (29)$$

Symptomatic patients (before being isolated or hospitalised:

$$iIS0_{s,a,,d+1,t+1} = (iIS0_{s,a,,d,t} - dthCov_{s,a,t} \cdot ds0_d) \cdot (1 - YS0_d) \quad (30)$$

The number of new (d=1) symptomatic patients being isolated/quarantined:

$$iSQ_{s,a,1,t} = SYM_{s,a,t} \cdot fSq_{a,t} \quad (31)$$

Isolated symptomatic patients:

$$iSQ_{s,a,d+1,t+1} = \sum_{d=1}^{60}(iSQ_{s,a,d,t} - dthCov_{s,a,t} \cdot dsq_d) \cdot (1 - YS1_d) \quad (32)$$

The number of new (d=1) symptomatic patients being hospitalised:

$$iSH_{s,a,1,t} = SYM_{s,a,t} \cdot fSh_{a,t} \quad (33)$$

Hospitalised symptomatic patients:

$$iSH_{s,a,d+1,t+1} = \sum_{d=1}^{60}(iSH_{s,a,d,t} - dthCov_{s,a,t} \cdot dsh_d) \cdot (1 - YH1_d) \quad (34)$$

The number newly recovered people (d=1):

$$iRE_{s,a,1,t+1} = \sum_{d=1}^{60}\left(iIA_{s,a,d,t} \cdot YA0_d + iIS0_{s,a,d,t} \cdot YS0_d + iISQ_{s,a,d,t} \cdot YS2_d + iISH_{s,a,d,t} \cdot YH2_d + VAC2_t\right) \quad (35)$$

Note: $VAC2_t$ is the number of newly vaccinated individuals who recovered from previous infections.

All recovered for d=1,2,3…tt:

$$iRE_{s,a,d+1,t+1} = (iRE_{s,a,d,t} - VAC2_t) \cdot (1 - mr_d) \cdot \left(1 - drOth_{s,a,t}\right) \quad (36)$$

All recovered individuals:

$$\boldsymbol{RE_{s,a,t}} = \sum_{d=1}^{tt} iRE_{s,a,d,t} \quad (37)$$

Derived reproduction values (R0, Rt)

The basic reproduction ratio (R0) is defined as the average number of individuals infected by a typical infectious individual in a total susceptible population, and effective reproduction ratio (Rt) is the number

Song F, Bachmann MO. *BMJ Open* 2021; 11:e053507. doi: 10.1136/bmjopen-2021-053507

of individuals infected by an infectious individual when only a proportion of the population are susceptible and the disease transmission dynamic may be influenced by control measures.[38] R values depend on the risk of infection per contact between an infectious and susceptible person, person-to-person contacts between individuals, the rate of transition from exposed to infectious, infectious period, and the prevalence of susceptible individuals in the population.[3] In this study, we don't use the reproduction ratio directly in simulating the spread of SARS-CoV-2 virus. To facilitate the understanding of effects of different intervention strategies, we estimated R0 and Rt during the simulation period, based on average values of relevant parameters and the calculation method used in a modelling study by Giordano et al.[27]

Average values of relevant parameters for estimating R values:

Weighted average fraction of symptomatic individuals in all infected individuals:

$$fS_t = \sum_a \left(\mu_a \cdot N_{3,a,t}/N_{3,11,t}\right) \tag{38}$$

Weighted average fraction of hospitalised symptomatic patients:

$$fH_t = \sum_a \left(fSH_{a,t} \cdot N_{3,a,t}/N_{3,11,t}\right) \tag{39}$$

Weighted average fraction of symptomatic patients self-isolated:

$$fQ_t = \sum_a \left(fSQq_{a,t} \cdot N_{3,a,t}/N_{3,11,t}\right) \tag{40}$$

Risk of daily transmission per infectious individual, depending on asymptomatic or symptomatic, household isolated or not:

$$\beta I0_t = \sum_{a,j=1}^{10} \mu_t \cdot Ca_{a,j,t} \cdot \frac{N_{j,t}}{N_{3,11,t}} \cdot (infA \cdot (1 - fS_t) + fS_t) \tag{41}$$

$$\beta IA_t = \sum_{a,j=1}^{10} \mu_t \cdot Ca_{a,j,t} \cdot \frac{N_{j,t}}{N_{3,11,t}} \cdot infA \cdot (1 - fS_t) \tag{42}$$

$$\beta SQ_t = \sum_{a,j=1}^{10} \mu_t \cdot Cb_{a,j,t} \cdot \frac{N_{j,t}}{N_{3,11,t}} \cdot (infA \cdot (1 - fS_t) + fS_t) \tag{43}$$

$$\beta S0_t = \sum_{a,j=1}^{10} \mu_t \cdot Ca_{a,j,t} \cdot \frac{N_{j,t}}{N_{3,11,t}} \tag{44}$$

The following transition variables are calculated for estimating R values:

$$d1_t = \alpha 2_t \cdot (1 - fS_t) \tag{45}$$

$$d2_t = \alpha 2_t \cdot fS_t \cdot fS0_t \tag{46}$$

$$d3_t = \alpha 2_t \cdot fS_t \cdot fSQ_t \tag{47}$$

$$d4_t = \alpha 2_t \cdot fS_t \cdot fSH_t \tag{48}$$

$$dA1_t = d1_t + d2_t + d3_t + d4_t \quad (49)$$

$$d5_t = \alpha 2_t \cdot (1 - fS_t \cdot e_2) \quad (50)$$

$$d6_t = \alpha 2_t \cdot fS_t \cdot fS0_t \cdot e_2 \quad (51)$$

$$d7_t = \alpha 2_t \cdot fS_t \cdot fSQ_t \cdot e_2 \quad (52)$$

$$d8_t = \alpha 2_t \cdot fS_t \cdot fSH_t \cdot e_2 \quad (53)$$

$$dA2_t = d5_t + d6_t + d7_t + d8_t \quad (54)$$

Effective reproductive value (Rt):

$$Rt = \frac{SU_{3,11,t}}{N_{3,11,t}} \left(\beta I0_t \cdot \frac{1}{dA1_t} + \beta IA_t \cdot \frac{d1_t}{dA1_t \cdot YA0_t} + \beta S0_t \cdot \frac{d2_t}{dA1_t \cdot YS0_t} + \beta SQ_t \cdot \frac{d3_t}{dA1_t \cdot YS1_t} + \beta SH_t \cdot \frac{d4_t}{dA1_t \cdot YH1_t}\right) + \left(\frac{SUre_{3,11,t}}{N_{3,11,t}} + \frac{VAC1_{3,11,t}}{N_{3,11,t}} \cdot e_1\right) \cdot \rho \cdot \left(\beta I0_t \cdot \frac{1}{dA2_t} + \beta IA_t \cdot \frac{d5_t}{dA2_t \cdot YA0_t} + \beta S0_t \cdot \frac{d6_t}{dA2_t \cdot YS0_t} + \beta SQ_t \cdot \frac{d7_t}{dA2_t \cdot YS1_t} + \beta SH_t \cdot \frac{d8_t}{dA2_t \cdot YH1_t}\right) \quad (55)$$

Song F, Bachmann MO. *BMJ Open* 2021; 11:e053507. doi: 10.1136/bmjopen-2021-053507

5. Modelling R code and input data files

R code used in this modelling study is provided below. Input data and parameters shown in the code is from a scenario with the following assumptions: annual revaccination during 2021-2024, long-term vaccine efficacy 62.5%, reduction in reinfectivity after waning of immunity 45%. Input data files required for running the R code are provided at the end of this section.

5.1 R code used

```
#########################################################################
## R code for simulating outbreak of covid-19 in England
#########################################################################

        tt <-1827               # 1827, from 1/1/20 to 31/12/24

#-------------------------------------------------------------------------------------------
                VAC <- 1            # Vaccination: 1- yes; 0-no
                VACtime <-367       # relevant if VAC=1
                back.nom <-1        # re-normality from 'back.tim', 1-Yes, 0-No
                back.tim <-566      # t=566:19/7/2021
                ASYinf <-0.21       # Relative infectivity of asymptomatic cases

                dGap <-365          # Interval between vac waves (day)

        #  VAC coverage for age groups
                vac.crg1 <-0.75             # 0.60    VAC coverage age 16-29
                vac.crg2 <-0.80             # 0.70    VAC coverage age 30-39
                vac.crg3 <-0.85             # 0.80    VAC coverage age 40-49
                vac.crg4 <-0.90             # 0.90    VAC coverage age 50+

                bta.nom <- 0.081    # from back.tim
   #++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

                 nRvac <-4                  # revaccination waves
                InfI2    <-0.45             # Relative infectivity after immunity loss
                rVEinf1  <-0.3876           # Efficacy of vaccine dose-1 on transmission risk
                rVEsym1  <-0.3876           # Efficacy of vaccine dose-1 on reduced symptomatic cases
                rVEinf2  <-0.6127           # Efficacy of vaccine dose-2 on transmission risk
                rVEsym2  <-0.6127           # Efficacy of vaccine dose-2 on reduced symptomatic cases
                rVEinf3  <-0.3876           # Long-term VE on infection
                rVEsym3  <-0.3876           # Long-term VE on symptoms
                rWinter1 <- 1.10            # Winter transmissibility: Sep-Oct, Mar-Apr
                rWinter2 <- 1.20            # Winter transmissibility: Nov-Feb
                pIM <-1                     # Natural immunity: 1-long-lived; 0-short-lived
                gwan.m <-730                # mean durability (days) of immunity
                gwan.s <-27.019             # gamma shape k
                pIMv <-1                    # Vaccine immunity 1:long-lived; 0:short-lived
                gwanv.m <-365               # mean durability (days) of vaccine immunity
                gwanv.s <-19.105            # gamma shape k


##++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
##  Vaccination timing, waves and age groups
##================================================================================
        vac.AGP  <- array(0, dim=c(7))              # no. of agegroups to be vaccinated phase 1-6
        vac.tim <- array(0, dim=c(18))              # starting time of each vaccination wave
        vagp       <- array(0, dim=c(18,7))     # agegroup (Vp:1-6, age 1,2)
#-------------------------------------------------------------------------------------------
        vac.tim[1] <-VACtime
        if(nRvac>1)                 {
        for(rev in 2:nRvac) {
          vac.tim[rev] <- vac.tim[rev-1] +dGap           # time of re-vaccination
                                    }
                                    }
```

```
        vac.dmaxi <- 180000         # Maximum no. of individuals vaccinated
 #-----------------------------------------------------------------------
                          vac.AGP[1] <-6            # 1: 70+;   2: 60-69;  3: 50-59;
                                                    # 4: 40-49; 5: 20-39;  6: 16(10)-19
        if(nRvac>1)                  {
        for(rev in 2:nRvac) {
                          vac.AGP[rev] <-6
                                     }
                                     }
 #-----------------------------------------------------------------------
 # Age ranges for vaccination from age phase 1 to 4
 #----------------------------------------------------------------------------
        vagp[1,1] <- 9               # age grp 9:       70+
        vagp[1,2] <- 10              # age grp 10:      85+
        vagp[2,1] <- 8               # age grp 8:       60-69
        vagp[2,2] <- 8               # age grp 8:
        vagp[3,1] <- 7               # age grp 7:       50-59
        vagp[3,2] <- 7               # age grp 7:
        vagp[4,1] <- 6               # age grp 6:       40-49
        vagp[4,2] <- 6               # age grp 6:
        vagp[5,1] <- 4               # age grp 4:       20-29
        vagp[5,2] <- 5               # age grp 5:       30-39
        vagp[6,1] <- 3               # age grp 3:       10-19
        vagp[6,2] <- 3               # age grp 3:       (40% for 16-19)
 #=====================================================================
        ts    <- 15           # Epidemic simulation start from 01/02/2020
        seeds <-1                    # Inital no. of exposed/infected
        dv0   <-14                   # Days required for developing immunity after vaccination
        tv0   <-63            # duration between dose-1 and dose-2: 9 weeks on average
        tw0   <-tt            # longest duration before immunity loss in recovered
        dd    <-60            # maximum days from exposed to recover or death.
        tm0   <-tt            # only used for defining death risk by days when tt<1096
        tage  <-367                  # t starting to shift age up

##$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
        beta       <- array(0, dim=c(tt))       # Infection risk per contact between susceptible and infected
        adjHsp     <- array(0, dim=c(tt))        # Adjust ratio for hospitalisation rate
        adjDth     <- array(0, dim=c(tt))        # Adjust case fatality to hospital fatality
###=============================================================================
##  Calibrate beta[t], adjHsp, adjDth
##==============================================================================
 for(t in 1:tt)        {              # No covid epidemdic before ts
          beta[t] <-0.094
 #--------------------------------------------
        if(NPI==1)            {
 #--------------------------------------------
        if(t>74)    {                # t> 15/03/2020
                beta[t] <- 0.069     # social distancing
                            }                #

        if(t>83)    {                # t> 23/03/2020
                beta[t] <-0.066      # Lockdown date
                            }                #

        if(t>132) {                  # t=132:11/05/2020
                beta[t] <-0.062      # Advised wearing face covering
                            }                #

        if(t>280) {                  # t=291: 17/10/2020:
                beta[t] <-0.077      # Increased transmissibility by new variant
                            }                #

        if(t>487) {                  # t>= 01/05/2021
                beta[t] <-0.081              # new delta variant
                            }                #

        if(back.nom==1)              {       # Returning to normal life
        if(t>(back.tim-1))    {
                          beta[t] <-bta.nom
                                     }
```


Song F, Bachmann MO. *BMJ Open* 2021; 11:e053507. doi: 10.1136/bmjopen-2021-053507

```
                                        }

                                }   # end if NPI==1

                        } # for t in ts:tt

#=============================================================================
#  Ratio for adjusting case fatality rate
#=============================================================================
            for(t in 1:tt)                  {
                                adjDth[t] <- 2.00               # higher death rate initially
            if(t>91)   {        adjDth[t] <- 1.80   }       # t=92:  1/4/20:
            if(t>100)  {        adjDth[t] <- 1.60   }
            if(t>110)  {        adjDth[t] <- 1.40   }
                                }   # end t in ts:tt
#=============================================================================
#  Ratio for calibrating hospitalisation rate
#=============================================================================
            for(t in 1:tt) {  adjHsp[t] <-0.30  }

##=============================================================================
## output files for simulation results
##-----------------------------------------------------------------------------
 Header.scen <-cbind("Time", "Rt", "POP_N", "SUS", "SUS.rec","SUS.vac", "RE", "VAef1", "VAef2", "New.VAC",
                     "New_Exp", "New_Sym","New_HS", "HS_pts", "Infect_sum", "DthCov-day", "DthCov_cum",
                     "DthAll_day" )   # End Header.out
  OFileName <- paste("...\\ResOut".out", sep="")
   write.table(Header.scen, file=OFileName, sep="\t", quote=FALSE,
                                        append=FALSE, row.names=FALSE, col.names=FALSE)
##=============================================================================
## Definition of Population and other key variables
##-----------------------------------------------------------------------------
            N    <- array(0, dim=c(tt,3,11))   # Total population[t,sex,age,cmb]
            SU   <- array(0, dim=c(tt,3,11))   # Suceptable[t,sex,age,cmb]
            EX   <- array(0, dim=c(tt,3,11))   # Exposed[t,sex,age,cmb]
            EXv1   <- array(0, dim=c(tt,3,11))   # Exposed in vaccinated
            EXv2   <- array(0, dim=c(tt,3,11))
            I0   <- array(0, dim=c(tt,3,11))   # Infected -presymptomatic
            I0v1   <- array(0, dim=c(tt,3,11))   # Infected -presymptomatic in vaccinated
            I0v2   <- array(0, dim=c(tt,3,11))
            IM0  <- array(0, dim=c(tt,3,11))   # Infected no quarantine -mild
            IM0v  <- array(0, dim=c(tt,3,11))   # Infection after vaccination or reinfection
            IS0  <- array(0, dim=c(tt,3,11))   # Infected no quarantine/no hospital -severe
            IS0v  <- array(0, dim=c(tt,3,11))   # Infection after vaccination or reinfection
            ISq  <- array(0, dim=c(tt,3,11))   # To be quarantined -severe
            ISqv  <- array(0, dim=c(tt,3,11))   # Infection after vaccination or reinfection
            ISh  <- array(0, dim=c(tt,3,11))   # To be hospitalised
            IShv  <- array(0, dim=c(tt,3,11))  # Infection after vaccination or reinfection
            QS  <- array(0, dim=c(tt,3,11))    # Quarantined -severe
            QSv  <- array(0, dim=c(tt,3,11))
            HS  <- array(0, dim=c(tt,3,11))   # Hospitalised -severe
            HSv  <- array(0, dim=c(tt,3,11))
            RE   <- array(0, dim=c(tt,3,11))   # Recovered

            eRE  <- array(0, dim=c(tt,tw0,3,11)) # By days since recovered/vacciated for waning immunity

            SU.vac   <- array(0, dim=c(tt,3,11))   # Suceptable from vaccinated
            SU.rec   <- array(0, dim=c(tt,3,11))   # Suceptable from recovered clinical
            vac.SU   <- array(0, dim=c(tt,3,11))   # Vaccinated suceptable[t,sex,age,cmb]
            vac.SUvac   <- array(0, dim=c(tt,3,11)) # vaccinated in loss of vaccine immunity
            vac.SUrec   <- array(0, dim=c(tt,3,11))  # vaccination of loss immunity in clinical recovered

            eEX   <- array(0, dim=c(tt,dd,3,11))   # Exposed[t,sex,age,cmb]
            eEXv1   <- array(0, dim=c(tt,dd,3,11))   # Exposed in vaccinated
            eEXv2   <- array(0, dim=c(tt,dd,3,11))
            eI0   <- array(0, dim=c(tt,dd,3,11))   # Infected -presymptomatic
            eI0v1   <- array(0, dim=c(tt,dd,3,11))   # Infected -presymptomatic in vaccinated
            eI0v2   <- array(0, dim=c(tt,dd,3,11))
            eIM0  <- array(0, dim=c(tt,dd,3,11))   # Infected no quarantine -mild
            eIM0v  <- array(0, dim=c(tt,dd,3,11))
```

```
eIS0  <- array(0, dim=c(tt,dd,3,11))   # Infected no quarantine/no hospital -severe
eIS0v  <- array(0, dim=c(tt,dd,3,11))
eISq  <- array(0, dim=c(tt,dd,3,11))   # To be quarantined -severe
eISqv  <- array(0, dim=c(tt,dd,3,11))
eISh  <- array(0, dim=c(tt,dd,3,11))   # To be hospitalised
eIShv  <- array(0, dim=c(tt,dd,3,11))
eQS  <- array(0, dim=c(tt,dd,3,11))   # Quarantined -severe
eQSv  <- array(0, dim=c(tt,dd,3,11))
eHS  <- array(0, dim=c(tt,dd,3,11))   # Hospitalised -severe
eHSv  <- array(0, dim=c(tt,dd,3,11))

 VA0   <- array(0, dim=c(tt,3,11))         # Vaccinated before immunity developed
 VAef1   <- array(0, dim=c(tt,3,11))       # Successful partial immunised
 VAef2   <- array(0, dim=c(tt,3,11))
eVA0    <- array(0, dim=c(tt,dv0,3,11)) # Specify days since vacciation 1-14 days
eVAef1   <- array(0, dim=c(tt,tv0,3,11))  # by days since vaccinated dose-1
eVAef2   <- array(0, dim=c(tt,tw0,3,11))  #  by days since vaccinated dose-2

eInf  <- array(0, dim=c(tt,dd,3,11))  #  Tracing with days since exposed/infected
eInfv  <- array(0, dim=c(tt,dd,3,11))  #  Tracing with days since exposed/infected in reinfected
Infect.sum <- array(0, dim=c(tt))          # Total no. of infected at t

#---------------------------------
# Temproary variable for up shifting ages

jN <- array(0, dim=c(10)); jSU <- array(0, dim=c(10)); jSU.vac <- array(0, dim=c(10))
jSU.rec <- array(0, dim=c(10)); jRE <- array(0, dim=c(10)); jEX <- array(0, dim=c(10))
jI0 <- array(0, dim=c(10)); jIM0 <- array(0, dim=c(10)); jIS0 <- array(0, dim=c(10))
jISq <- array(0, dim=c(10)); jISh <- array(0, dim=c(10)); jQS <- array(0, dim=c(10))
jHS <- array(0, dim=c(10))

jEXv1 <- array(0, dim=c(10)); jI0v1 <- array(0, dim=c(10))
jEXv2 <- array(0, dim=c(10)); jI0v2 <- array(0, dim=c(10)); jIM0v <- array(0, dim=c(10))
jIS0v <- array(0, dim=c(10)); jISqv <- array(0, dim=c(10)); jIShv <- array(0, dim=c(10))
jQSv <- array(0, dim=c(10)); jHSv <- array(0, dim=c(10)); jVA0 <- array(0, dim=c(10))
jVAef1 <- array(0, dim=c(10)); jVAef2 <- array(0, dim=c(10))

jeEX <- array(0, dim=c(dd,10)); jeI0 <- array(0, dim=c(dd,10)); jeIM0 <- array(0, dim=c(dd,10))
jeIS0 <- array(0, dim=c(dd,10)); jeISq <- array(0, dim=c(dd,10)); jeISh <- array(0, dim=c(dd,10))
jeQS <- array(0, dim=c(dd,10)); jeHS <- array(0, dim=c(dd,10))

jeEXv1 <- array(0, dim=c(dd,10)); jeI0v1 <- array(0, dim=c(dd,10)); jeEXv2 <- array(0, dim=c(dd,10))
jeI0v2 <- array(0, dim=c(dd,10))

jeIM0v <- array(0, dim=c(dd,10)); jeIS0v <- array(0, dim=c(dd,10)); jeISqv <- array(0, dim=c(dd,10))
jeIShv <- array(0, dim=c(dd,10)); jeQSv <- array(0, dim=c(dd,10)); jeHSv <- array(0, dim=c(dd,10))
jeInf  <- array(0, dim=c(dd,10)); jeInfv  <- array(0, dim=c(dd,10))

jeVA0    <- array(0, dim=c(dv0,10)); jeVAef1   <- array(0, dim=c(tv0,10)); jeVAef2   <- array(0, dim=c(tw0,10))
jeRE  <- array(0, dim=c(tw0,10))

#-----------------------------------

HS.new  <- array(0, dim=c(tt,3,11)) ; allHS  <- array(0, dim=c(tt))

Bsex   <- array(0, dim=c(2))  # % sex ratio for births
dBth   <- array(0, dim=c(11))              # No. of births/day, note: dBth[>1] <-0

I.i0  <- array(0, dim=c(tt,11))            # total no. of pre-clin infectious
I.m0  <- array(0, dim=c(tt,11))            # total no. of mild cases - not quarantined
I.m  <- array(0, dim=c(tt,11))
I.s0  <- array(0, dim=c(tt,11))            # total no. of severe cases - not quarantined
I.s  <- array(0, dim=c(tt,11))
I.sq  <- array(0, dim=c(tt,11))            # total no. of severe cases - quarantined
I.i0v  <- array(0, dim=c(tt,11))           # total no. of pre-clin infectious
I.m0v  <- array(0, dim=c(tt,11))           # total no. of mild cases - not quarantined
I.mv  <- array(0, dim=c(tt,11))
I.s0v  <- array(0, dim=c(tt,11))           # total no. of severe cases - not quarantined
I.sv  <- array(0, dim=c(tt,11))
I.sqv  <- array(0, dim=c(tt,11))           # total no. of severe cases - quarantined
```

```
IAl   <- array(0, dim=c(tt))

Rt  <- array(0, dim=c(tt))
exposed.n  <- array(0, dim=c(tt))

drisk.inf  <- array(0, dim=c(2,11))  # death risk in symptomatic cases
rHosp <- array(0, dim=c(11))       # Case hospitalisation rate

rADth0  <- array(0, dim=c(366,3,11))  # 5yr average death rate/1000/day without comordity

 nDth.inf <- array(0, dim=c(tt,3,11))  # Covid related deaths -all
nDth.s0 <- array(0, dim=c(tt,dd,3,11))  # Covid related deaths -infected
nDth.sq <- array(0, dim=c(tt,dd,3,11))  # Covid related deaths -infected
nDth.ss <- array(0, dim=c(tt,3,11))  # Covid related deaths -outside hospital
nDth.hos <- array(0, dim=c(tt,dd,3,11))  # Covid related deaths-hospital

 nDth.infv <- array(0, dim=c(tt,3,11))  # Covid related deaths -all
nDth.s0v <- array(0, dim=c(tt,dd,3,11))  # Covid related deaths -infected
nDth.sqv <- array(0, dim=c(tt,dd,3,11))  # Covid related deaths -infected
nDth.ssv <- array(0, dim=c(tt,3,11))  # Covid related deaths -outside hospital
nDth.hosv <- array(0, dim=c(tt,dd,3,11))  # Covid related deaths-hospital

nDth.cov <- array(0, dim=c(tt,3,11))  # Covid related deaths
nDth.oth  <- array(0, dim=c(tt,3,11))  # no. of other deaths in SU, RE and VAC

nDth.all <- array(0, dim=c(tt,3,11))  # All cause deaths including covid & non-covid deaths
nDthHos.all <- array(0, dim=c(tt,3,11))  # All cause deaths including covid & non-covid deaths
nDthHos.allv <- array(0, dim=c(tt,3,11))

nDthcum.cov <- array(0, dim=c(tt))       # Covid related deaths -Cumulative

beta2.i0 <- array(0, dim=c(tt)); beta2.m0 <- array(0, dim=c(tt)); beta2.mq <- array(0, dim=c(tt))
beta2.s0 <- array(0, dim=c(tt)); beta2.sq <- array(0, dim=c(tt)); beta2.al <- array(0, dim=c(tt))

dk <- array(0, dim=c(tt,18)); dA  <- array(0, dim=c(tt)); dAv1 <- array(0, dim=c(tt))
dAv2 <- array(0, dim=c(tt))

gdth.inf <- array(0, dim=c(dd))          # gamma distribution of deaths by days since infected
gdth.hos <- array(0, dim=c(dd))          # gamma distribution of deaths by days since hos admin
alp1     <- array(0, dim=c(dd))          # Transition rate from exposed to I0
alp2     <- array(0, dim=c(dd))          # Transition rate from I0 to I_clinical
gm0.m    <- array(0, dim=c(tt))          # gamma mean from IM0 to RE
gm0.s    <- array(0, dim=c(tt))          # gamma sd
gs0.m    <- array(0, dim=c(tt))          # gamma mean from IS0 to RE
gs0.s    <- array(0, dim=c(tt))          # gamma sd
gs1.m    <- array(0, dim=c(tt))          # gamma mean from ISq to QS
gs1.s    <- array(0, dim=c(tt))          # gamma sd
gs2.m    <- array(0, dim=c(tt))          # gamma mean from QS to RE
gs2.s    <- array(0, dim=c(tt))          # gamma sd
gh1.m    <- array(0, dim=c(tt))          # gamma mean from ISh to HS
gh1.s    <- array(0, dim=c(tt))          # gamma sd
gh2.m    <- array(0, dim=c(tt))          # gamma mean from HS to RE
gh2.s    <- array(0, dim=c(tt))          # gamma sd

# Day since infection specific transition rates:
gam.m0   <- array(0, dim=c(tt,dd))       # duration from IM0 to RE
gam.s0   <- array(0, dim=c(tt,dd))       # duration from IS0 to RE
gam.s1   <- array(0, dim=c(tt,dd))       # duration from ISq to QS
gam.s2   <- array(0, dim=c(tt,dd))       # duration from QS to RE
gam.h1   <- array(0, dim=c(tt,dd))       # duration from ISh to HS
gam.h2   <- array(0, dim=c(tt,dd))       # duration from HS to RE

# Average transition rates for estimating Rt
gam.m0a <- array(0, dim=c(tt))           # duration from IM0 to RE
gam.s0a  <- array(0, dim=c(tt))          # duration from IS0 to RE
gam.s1a  <- array(0, dim=c(tt))          # duration from ISq to QS
gam.s2a  <- array(0, dim=c(tt))          # duration from QS to RE
gam.h1a  <- array(0, dim=c(tt))          # duration from ISh to HS
gam.h2a  <- array(0, dim=c(tt))          # duration from HS to RE
```

```
gimm.wan <- array(0, dim=c(tw0))  # Distribution of immunity waning
gimmv.wan <- array(0, dim=c(tw0))  # Distribution of vaccine immunity waning

cum_gdth.inf <- array(0, dim=c(dd))     # gamma distribution of deaths by days since infected
cum_gdth.hos <- array(0, dim=c(dd))     # gamma distribution of deaths by days since hos admin
cum_alp1 <- array(0, dim=c(dd))         # Transition rate from exposed to I0
cum_alp2 <- array(0, dim=c(dd))         # Transition rate from I0 to I_clinical
cum_gm0.m       <- array(0, dim=c(tt))          # gamma mean from IM0 to RE
cum_gm0.s       <- array(0, dim=c(tt))          # gamma sd
cum_gs0.m       <- array(0, dim=c(tt))          # gamma mean from IS0 to RE
cum_gs0.s       <- array(0, dim=c(tt))          # gamma sd
cum_gs1.m       <- array(0, dim=c(tt))          # gamma mean from ISq to QS
cum_gs1.s       <- array(0, dim=c(tt))          # gamma sd
cum_gs2.m       <- array(0, dim=c(tt))          # gamma mean from QS to RE
cum_gs2.s       <- array(0, dim=c(tt))          # gamma sd
cum_gh1.m       <- array(0, dim=c(tt))          # gamma mean from ISh to HS
cum_gh1.s       <- array(0, dim=c(tt))          # gamma sd
cum_gh2.m       <- array(0, dim=c(tt))          # gamma mean from HS to RE
cum_gh2.s       <- array(0, dim=c(tt))          # gamma sd

# Day since infection specific transition rates:
cum_gam.m0      <- array(0, dim=c(tt,dd))       # duration from IM0 to RE
cum_gam.s0      <- array(0, dim=c(tt,dd))       # duration from IS0 to RE
cum_gam.s1      <- array(0, dim=c(tt,dd))       # duration from ISq to QS
cum_gam.s2      <- array(0, dim=c(tt,dd))       # duration from QS to RE
cum_gam.h1      <- array(0, dim=c(tt,dd))       # duration from ISh to HS
cum_gam.h2      <- array(0, dim=c(tt,dd))       # duration from HS to RE

# Average transition rates for estimating Rt
cum_gam.m0a     <- array(0, dim=c(tt))          # duration from IM0 to RE
cum_gam.s0a     <- array(0, dim=c(tt))          # duration from IS0 to RE
cum_gam.s1a     <- array(0, dim=c(tt))          # duration from ISq to QS
cum_gam.s2a     <- array(0, dim=c(tt))          # duration from QS to RE
cum_gam.h1a     <- array(0, dim=c(tt))          # duration from ISh to HS
cum_gam.h2a     <- array(0, dim=c(tt))          # duration from HS to RE

cum_gimm.wan <- array(0, dim=c(tw0))  # Distribution of immunity waning
cum_gimmv.wan <- array(0, dim=c(tw0))  # Distribution of vaccine immunity waning

rFm     <- array(0, dim=c(11))                  # % from I0 to Mild -age-specific
rFsq    <- array(0, dim=c(tt,11))       # % from I0 to QS
rFhos   <- array(0, dim=c(tt,11))       # % from I0 to HS

rFhos.al  <- array(0, dim=c(tt)); rFm.al  <- array(0, dim=c(tt)); rFsq.al  <- array(0, dim=c(tt))

New.expsu   <- array(0, dim=c(tt,3,11))# Newly exposed in SU
New.expv0  <- array(0, dim=c(tt,3,11)) # Newly exposed in VA0
New.expv1  <- array(0, dim=c(tt,3,11)) # Newly exposed in VAef1
New.expv2  <- array(0, dim=c(tt,3,11)) # Newly exposed in VAef2
New.expal  <- array(0, dim=c(tt,3,11))  # Newly exposed -all

Expv0     <- array(0, dim=c(tt,dv0,3,11))       # Newly exposed in VA0 by day
Expv1     <- array(0, dim=c(tt,tv0,3,11))       # Newly exposed in VAef by day
Expv2     <- array(0, dim=c(tt,tt,3,11))

New.explosv1  <- array(0, dim=c(tt,3,11))       # Newly exposed due to loss immunity in vaccinated
New.explosrec <- array(0, dim=c(tt,3,11))       # Newly exposed due to loss immunity in recovered

New.vac   <- array(0, dim=c(tt,3,11))    # Newly vaccinated

New.sym   <- array(0, dim=c(tt,3,11))   # New symptomatic

eSeed <- array(0, dim=c(tt,dd,3,11))   # Seeds of exposed
Seed.all <- array(0, dim=c(tt,3,11))   # Seeds of exposed

CMTa <- array(0, dim=c(11,11))     # Contact matrix by age for all contacts symmetric
CMTh <- array(0, dim=c(11,11))     # Contact matrix by age for home contacts symmetric

adjCNTa <- array(0, dim=c(tt,11))     # adjust no. of Contacts at home by age & m
```

```
          V.day      <- array(0, dim=c(18,7))          # no. of days for vaccination (age grp 1-6)
          V.dmax     <- array(0, dim=c(18,7))          # maximum vaccinated per day by age grp (1-6)
          T.vac      <- array(0, dim=c(18,7))          # maximum vaccinated per day by age grpe (1-6)
          vacend     <- array(0, dim=c(18,7))          # End date for vaccinating group Vp

## READ: data on initial population 01-01-2020

 inPOP <-read.table(file="...\\inPOPEng20.csv", header=TRUE, sep=",")

# Obtain data on initial population in England 2020 (start of the year)

          for(a in 1:10) {    # Input UK population

           N[1,1,a] <-inPOP[a,3]*1000          #   male all
           N[1,2,a] <-inPOP[a,4]*1000          # female all

          drisk.inf[1,a] <-inPOP[a,5]          # case fatality   male
          drisk.inf[2,a] <-inPOP[a,6]          # case fatality female

          rHosp[a] <- inPOP[a,7]  # Hospitalisation rate (cases)

                              }   # End input POP data

  #---------------------------------------------------
          for(s in 1:2)          {
          for(a in 1:10)         {
            N[1,3,11] <- N[1,3,11] + N[1,s,a];  N[1,s,11] <- N[1,s,11] + N[1,s,a];  N[1,3,a] <- N[1,3,a] + N[1,s,a]
           SU[1,s,a] <-  N[1,s,a];  SU[1,3,11] <-SU[1,3,11] +SU[1,s,a]

                              }
                              }

##================================================
##   Birth related parameters, note: CMB[s,a]
##================================================
          dBth[1] <-1800              # No. of births per day; for day-1 only
                                      # dBth =no. of deaths/day for other days
          Bsex[1] <-0.512             # male %
          Bsex[2] <-0.488             # female %

## READ: data on 5 year (2015-2019) average death rate/1000/day

 inAllDth <-read.table(file="...\\inDTH1519Eng10.csv", header=TRUE, sep=",")
                    d<- 1             # day from 1 to 364=7x52
          for(w in 1:52) {      # Input death rate for male week 1-52
          for(dw in 1:7) {
           for(a in 1:10) {
                    cc<-a+1
           rADth0[d,1,a] <-inAllDth[w,cc]/1000
                               }          # end a in 1:10
                      d <- d+1
                              }   # end for dw in 1:7
                              }   # end for w  in 1:52
                    d<- 1             # day from 1 to 364=7x52
          for(w in 53:104) {          # Input death rate for female week 1-52
          for(dw in 1:7) {
           for(a in 1:10)  {
                    cc<-a+1
           rADth0[d,2,a] <-inAllDth[w,cc]/1000
                                }     # end a in 1:10
                      d <- d+1
                              }
                              }

          for(s in 1:2)          {
          for(a in 1:10)         {
          rADth0[365,s,a] <- rADth0[364,s,a]   # 366 days in 2020
          rADth0[366,s,a] <- rADth0[364,s,a]   # 366 days in 2020
                              }
```

```
                                    }

## READ: contact matrix all and at home
                inCM <-read.table(file="…\\inCMATRIX10.csv", header=TRUE, sep=",")

     for(a in 1:10)     {    # Input Contact matrix data
     for(j in 1:10)     {
      # All contacts (for community transmission)
                                    ca <-j+1
                          CMTa[a,j] <-inCM[a,ca]
      # Contacts at home (for home isolation)
                                    ch <-j+11
                          CMTh[a,j] <-inCM[a,ch]
                                    }
                                    }

##-----------------------------------------------------
## Key input parameters
##-----------------------------------------------------
    # % of mild illness among infected - age-specific, constant overtime
    # Data source: Davies et al Nature Med: Age-dependent effects in transmission
              for(a in 1:2)            { rFm[a] <- 0.71 }       # age 0-9
                                         rFm[3] <- 0.79         # age 10-19
                                         rFm[4] <- 0.73         # age 20-29
                                         rFm[5] <- 0.67         # age 30-39
                                         rFm[6] <- 0.60         # age 40-49
                                         rFm[7] <- 0.51         # age 50-59
                                         rFm[8] <- 0.37         # age 60-69
              for(a in 9:10)  { rFm[a] <- 0.31 }       # age 70+

  #-------------------------------------------------
  #          Verity et al:
  #          17.8 days from onset to deaths

              gdth.m <-23.0         # mean days from exposed to deaths
              gdth.s <-4.796        # sd

#             READ: data on other input parameters

  inParam <-read.table(file="...\\inParamet20.csv", header=TRUE, sep=",")

   for(t in ts:tt)        {      # Epidemic simulation starts from ts
              if(t>(ts-1))              {             ### t>ts
                          c <-2                     # data column
                                    }

   #=================================================================
   #  Effects of NPI control measures
   #-------------------------------------------------------------------
   # Control-1: case based self-isolation mandated on 12/03/2020
   #  Reduced infectious periods for symptomatic patients since 13/03/2020
   #-------------------------------------------------------------------
              if(t>72)  {    #  after date: 12/03/2020 -self isolation of symptomatic individuals
                          c <-3
                            }
   #-------------------------------------------------------------------
   # Control-2:  Social distance encouraged in the UK 16/03/2020
   #  shielding of vulnerable people
   #-------------------------------------------------------------------
              if(t>76)  {     # After date 16/03/2020 -social distance
                          c <-4
                            }
   #=================================================================
   # Control-3: lockdown from 24/03/2020 in the UK; stay at home and other restrictions
   #-------------------------------------------------------------------
              if(t>83)  {             # lockdown initial
                          c <-5
                            }
   #=================================================================
   # On 4 July 2020: third step in easing national restrictions
```

```
#  including 2m => 2m or 1m social distancing, restaurants/pubs
#   reopening, holiday accommodations, museum, place of worships, ...
#=============================================================
        if(t>186) {             # date: 04/07/2020 more shops/pubs open
                        c <-6
                                }
 #=====================================================
 # Reopening of schools from September 2020
 #=====================================================
        if(t>244) {             # date: 01/09/2020 more shops/pubs re-opening
                        c <-7
                                }
#=====================================================
# 2nd national restrictions
#-----------------------------------------------------
        if(t>309) {             # Date: 05/11/2020 localised control measures
                        c <-8
                                 }
#=====================================================
# Lifting national restrictions
#-----------------------------------------------------
        if(t>336) {             # Date: 02/12/2020 Lifing national restrictions
                        c <-9
                                 }
#=====================================================
# National lockdown restrictions
#-----------------------------------------------------
        if(t>370) {             # Date: 05/01/2021 National restrictions
                        c <-10
                                 }
#=====================================================
# Partial lifting national lockdown restrictions
#-----------------------------------------------------
        if(t>432) {             #  t=440: 15/03/2021; partially lifting national restrictions
                        c <-11
                                 }
        if(t>505) {             #  t=506: 20/05/2021
                        c <-12
                                 }
#=====================================================
# Return to normal from back.tim
#-----------------------------------------------------
        if(back.nom==1)    {
        if(t>(back.tim-1))   {
                        c <-13
                                 }
                                 }

# ### Get data from dataframe "inPar"
        # % home quarantine among symptomatic cases

        for(a in 1:6)           {  # age 0-49
                rFsq[t,a] <- inParam[1,c]
                                }
        for(a in 7:8)   {  # age 50-69
                rFsq[t,a] <- inParam[2,c]
                                }
        for(a in 9:10)  {  # age 70+
                rFsq[t,a] <- inParam[3,c]
                                }

 #-----------------------------------------------------------
 # Transition parameters for E-I0-TM/S-QM/S-RE
 #  gamma distribution mean and shape
 #-----------------------------------------------------------

        alp1.m <- inParam[4,c]          # gamma mean -days incupation non-infectious period
        alp1.s <- inParam[5,c]          # gamma sd
        alp2.m <- inParam[6,c]          # gamma mean -infectiou duration before symptomatic
        alp2.s <- inParam[7,c]          # gamma sd
```

```
            gm0.m[t] <-inParam[8,c]      # mean days infectious period- no quarantine -mild
            gm0.s[t] <-inParam[9,c]      # gamma sd

            gs0.m[t] <-inParam[10,c]     # mean duration of infectious no quarantine -severe
            gs0.s[t] <-inParam[11,c]     # gamma sd

            gs1.m[t] <-inParam[12,c]     # mean days of infectious before quarantine -severe
            gs1.s[t] <-inParam[13,c]     # gamma sd
            gs2.m[t] <-inParam[14,c]     # mean duration from quarantine to recovery -severe
            gs2.s[t] <-inParam[15,c]     # gamma sd

            gh1.m[t] <-inParam[16,c]     # mean days before hospitalisation -severe
            gh1.s[t] <-inParam[17,c]     # gamma sd
            gh2.m[t] <-inParam[18,c]     # mean days of non-ICU hospitalisation -severe
            gh2.s[t] <-inParam[19,c]     # gamma sd

  #-------------------------------------------------
  #  adjCNT=1, no change; adjCNT<1, reduced contacts
  #-------------------------------------------------
            for(a in 1:3)                {          # Age 0-19 children
                  adjCNTa[t,a] <- inParam[20,c]
                                         }
            for(a in 4:7)                {          # Age 20-59
                  adjCNTa[t,a] <- inParam[21,c]
                                         }
            for(a in 8:8)                {          # Age 60-69
                  adjCNTa[t,a] <- inParam[22,c]
                                         }
            for(a in 9:10)       {          # Age 70+
                  adjCNTa[t,a] <- inParam[23,c]
                                         }

                              }          ## End tt for input parameters

#-------------------------------------------------------------------------
    # Hospitalisation rate by age grp based on Verity et al.
            for(t in 1:tt)       {
            for(a in 1:10)       {
             rFhos[t,a] <-(rHosp[a]/(1-rFm[a])) *adjHsp[t]
                              }
                              }
#=========================================================================================
  #  Calculating transition rate by days since exposed, infected, ...
  #   cumulative (pgamma) from 1 to dd: eventually all will be transfered to next status ....
  #================================================================ ====
            for(d in 1:dd)       {
                cum_alp1[d] <- pgamma(d, (alp1.m/alp1.s)^2, scale=(alp1.s^2)/alp1.m, log=FALSE)
                cum_alp2[d] <- pgamma(d, (alp2.m/alp2.s)^2, scale=(alp2.s^2)/alp2.m, log=FALSE)
             cum_gdth.inf[d] <- pgamma(d, (gdth.m/gdth.s)^2, scale=(gdth.s^2)/gdth.m, log=FALSE)
                              }

            # Translating cumulative probability to transition rate by d
                     alp1[1] <- cum_alp1[1]
                    alp2[1] <- cum_alp2[1]
                  gdth.inf[1] <- cum_gdth.inf[1
            for(d in 2:dd)       {
                  if(cum_alp1[d-1]<1) {
              alp1[d] <- (cum_alp1[d]-cum_alp1[d-1])/(1-cum_alp1[d-1])
                                         } else {    alp1[d] <-1             }

                  if(cum_alp2[d-1]<1) {
              alp2[d] <- (cum_alp2[d]-cum_alp2[d-1])/(1-cum_alp2[d-1])
                                         } else {    alp2[d] <-1             }

                  if(cum_gdth.inf[d-1]<1)       {
             gdth.inf[d] <- (cum_gdth.inf[d]-cum_gdth.inf[d-1])/(1-cum_gdth.inf[d-1])
                                         } else {    gdth.inf[d]<-1        }

                              }
```

```
        for(t in ts:tt)         {       # start t in 1:tt

        for(d in 1:dd)          {       # start d in 1:dd
        cum_gam.m0[t,d] <- pgamma(d, (gm0.m[t]/gm0.s[t])^2, scale=gm0.s[t]^2/gm0.m[t], log=FALSE)

        cum_gam.s0[t,d] <- pgamma(d, (gs0.m[t]/gs0.s[t])^2, scale=gs0.s[t]^2/gs0.m[t], log=FALSE)
        cum_gam.s1[t,d] <- pgamma(d, (gs1.m[t]/gs1.s[t])^2, scale=gs1.s[t]^2/gs1.m[t], log=FALSE)
        cum_gam.s2[t,d] <- pgamma(d, (gs2.m[t]/gs2.s[t])^2, scale=gs2.s[t]^2/gs2.m[t], log=FALSE)

        cum_gam.h1[t,d] <- pgamma(d, (gh1.m[t]/gh1.s[t])^2, scale=gh1.s[t]^2/gh1.m[t], log=FALSE)
        cum_gam.h2[t,d] <- pgamma(d, (gh2.m[t]/gh2.s[t])^2, scale=gh2.s[t]^2/gh2.m[t], log=FALSE)
                        }   # end d in 1:dd

        gam.m0[t,1] <- cum_gam.m0[t,1]; gam.s0[t,1] <- cum_gam.s0[t,1]; gam.s1[t,1] <- cum_gam.s1[t,1]
        gam.s2[t,1] <- cum_gam.s2[t,1]; gam.h1[t,1] <- cum_gam.h1[t,1]; gam.h2[t,1] <- cum_gam.h2[t,1]

        for(d in 2:dd)          {       # start d in 2:dd

                if(cum_gam.m0[t,d-1]<1)     {
        gam.m0[t,d] <- (cum_gam.m0[t,d]-cum_gam.m0[t,d-1])/(1-cum_gam.m0[t,d-1])
                                    } else {    gam.m0[t,d]<-1      }
                if(cum_gam.s0[t,d-1]<1)     {
        gam.s0[t,d] <- (cum_gam.s0[t,d]-cum_gam.s0[t,d-1])/(1-cum_gam.s0[t,d-1])
                                    } else {    gam.s0[t,d]<-1      }
                if(cum_gam.s1[t,d-1]<1)     {
        gam.s1[t,d] <- (cum_gam.s1[t,d]-cum_gam.s1[t,d-1])/(1-cum_gam.s1[t,d-1])
                                    } else {    gam.s1[t,d]<-1      }
                if(cum_gam.s2[t,d-1]<1)     {
        gam.s2[t,d] <- (cum_gam.s2[t,d]-cum_gam.s2[t,d-1])/(1-cum_gam.s2[t,d-1])
                                    } else {    gam.s2[t,d]<-1      }
                if(cum_gam.h1[t,d-1]<1)     {
        gam.h1[t,d] <- (cum_gam.h1[t,d]-cum_gam.h1[t,d-1])/(1-cum_gam.h1[t,d-1])
                                    } else {    gam.h1[t,d]<-1      }
                if(cum_gam.h2[t,d-1]<1)     {
        gam.h2[t,d] <- (cum_gam.h2[t,d]-cum_gam.h2[t,d-1])/(1-cum_gam.h2[t,d-1])
                                    } else {    gam.h2[t,d]<-1      }
                                    }   # end d in 2:dd

        #---------------------------------------------------------------------------
        #  Average transition rates for estimating Rt.
        #   Average transition rates were estimated by exponential distribution,
        #---------------------------------------------------------------------------
                alp2.a    <-(1/alp2.m)/1.337            # gamma m=1.5
                gam.m0a[t] <-(1/gm0.m[t])/1.11                  # gamma m=5
                gam.s0a[t] <-(1/gs0.m[t])/1.11                  # gamma m=5
                gam.s1a[t] <-(1/gs1.m[t])/1.25                  # gamma m=2
                gam.s2a[t] <-(1/gs2.m[t])/1.125                 # gamma m=4
                gam.h1a[t] <-(1/gh1.m[t])/1.25                  # gamma m=2
                        }       # end t in 1:tt

#-------------------------------------------------------------------------------
# Distribution of immunity duration:
        for(w in 1:tw0)     {       # start w in 1:tw
        cum_gimm.wan[w] <- pgamma(w, (gwan.m/gwan.s)^2, scale=gwan.s^2/gwan.m, log=FALSE)
                        }       # end w in 1:tw

                gimm.wan[1] <- cum_gimm.wan[1]

        for(w in 2:tw0)     {
                if(cum_gimm.wan[w-1]<1) {
        gimm.wan[w] <- (cum_gimm.wan[w]-cum_gimm.wan[w-1])/(1-cum_gimm.wan[w-1])
                          } else { gimm.wan[w]<-1    }
                        }

  if(pIM==1)    {   # If permanent immunity -gimm.wan=0
        for(w in 1:tw0)     {
                gimm.wan[w] <- 0.0
                        }
                        }   # end if pIM=1
```

```
  # Distribution of vaccine immunity duration:

        for(w in 1:tw0)       {             # start w in 1:tw
        cum_gimmv.wan[w] <- pgamma(w, (gwanv.m/gwanv.s)^2, scale=gwanv.s^2/gwanv.m, log=FALSE)
                              }             # end w in 1:tw

                gimmv.wan[1] <- cum_gimmv.wan[1]

        for(w in 2:tw0)       {
                if(cum_gimmv.wan[w-1]<1) {
        gimmv.wan[w] <- (cum_gimmv.wan[w]-cum_gimmv.wan[w-1])/(1-cum_gimmv.wan[w-1])
                                }       else        {       gimmv.wan[w] <-1 }
                              }

    if(pIMv==1)     {   # If permanent vaccine immunity -gimm.wan=0
        for(w in 1:tw0)       {
                gimmv.wan[w] <- 0.0
                              }
                              }   # end if pIMv=1


##============================================================
##  Simulation of covid-19 epidemic
##============================================================
   for(t in 2:tt)       {            ############ Start simulation
#====================================================================
# Seasonality adjustment of beta[t]

        if(t>639) {             # From Sept 2021
                beta[t]<- bta.nom
        if(t.dr >240)           {                       # from about Sept
        if(t.dr <301)           {                       # to about Oct
        beta[t]<- bta.nom *rWinter1  # beta increased by rWinter
                              }
                              }
        if(t.dr >300)           {                       # from Nov to Dec
        beta[t]<- bta.nom *rWinter2  # beta increased by rWinter
                              }
        if(t.dr <60)            {                       # From Jan to Feb
        beta[t]<- bta.nom *rWinter2  # beta increased by rWinter
                              }
        if(t.dr >59)            {                       # from Mar
        if(t.dr <120)           {                       # to about Apr
        beta[t]<- bta.nom *rWinter1  # beta increased by rWinter
                              }
                              }
                              }   # end t>639

#=======================================================
#  Days of the year for non-covid death rate
#=======================================================
        if(t==2)   {   t.dr <-1             }           # start year 1
        if(t==368)             {   t.dr <-1             }           # start year 2
        if(t==733)             {   t.dr <-1             }           # start year 3
        if(t==1098)            {   t.dr <-1             }           # start year 4
        if(t==1463)            {   t.dr <-1             }           # start year 5
        if(t==1828)            {   t.dr <-1             }           # start year 6
        if(t==2193)            {   t.dr <-1             }           # start year 7
        if(t==2558)            {   t.dr <-1             }           # start year 8
        if(t==2923)            {   t.dr <-1             }           # start year 9
        if(t==3288)            {   t.dr <-1             }           # start year 10

#====================================================================
#  First running through t from 1 to ts a period of no covid epidemic

    if(t<ts)                    {    #####  The susceptile from t=1 to t=ts
        for(a in 1:10)          {
        for(s in 1:2)           {
          N[t,s,a] <- N[t-1,s,a]*(1-rADth0[t.dr,s,a]) +dBth[a]*Bsex[s];  SU[t,s,a] <- N[t,s,a]
```

```
                                } }
                                }   #### end if t<ts

#=================================================================
#  Simulating from ts to tt for covid epidemic
#-----------------------------------------------------------------------------------
            if(t==ts)   {
                        eSeed[ts,3,1,5] <- seeds
            for(a in 1:10)          {
            for(s in 1:2)           {
            for(m in 1:2)           {
              N[t,s,a] <-N[t-1,s,a]*(1-rADth0[t.dr,s,a]) +dBth[a]*Bsex[s];  SU[t,s,a] <-N[t,s,a]
                        for(d in 1:10)          {
             Seed.all[t,s,a] <-Seed.all[t,s,a] +eSeed[t,d,s,a]
                                        }
                                } } }
                                }  # end if t==ts

#=====================================================================
            if(t>ts)    {    #### start simulating covid epidemic

#  Seeding exposed per day from 15/01/2020 to 29/02/2020 (t=60)
             if(t<40) {  eSeed[t,3,1,5] <- eSeed[t-1,3,1,5] +1    }

            for(a in 1:10)          {
            for(s in 1:2)           {
            for(d in 1:10)          {
             Seed.all[t,s,a] <-Seed.all[t,s,a] +eSeed[t,d,s,a]
                                }
                                }
                                }

  #=========================================================
  # Average rFm & rFhos for estimating R0/Rt: weighted by age specific proportion of relevant cases

            for(a in 1:10)          {
                        rFsq.al[t-1] <- rFsq.al[t-1]+  rFsq[t-1,a]*N[t-1,3,a]/N[t-1,3,11]
                         rFm.al[t-1] <-  rFm.al[t-1] +      rFm[a]*N[t-1,3,a]/N[t-1,3,11]
                       rFhos.al[t-1] <-rFhos.al[t-1] +rFhos[t-1,a]*N[t-1,3,a]/N[t-1,3,11]
                                }  # end a in 1:10

#=============================================================================
# Overall infectious individuals in the population

            for(a in 1:10)          {
            for(s in 1:2)           {
           I.i0[t-1,a] <-I.i0[t-1,a] +I0[t-1,s,a]                                   # pre-clinical infectious
           I.i0v[t-1,a] <-I.i0v[t-1,a] +I0v1[t-1,s,a] +I0v2[t-1,s,a]
            I.m0[t-1,a] <- I.m0[t-1,a] +IM0[t-1,s,a]
             I.m[t-1,a] <-  I.m[t-1,a] +IM0[t-1,s,a] +I0[t-1,s,a]*rFm[a]  # All asymptomatic/mild cases
           I.m0v[t-1,a] <-I.m0v[t-1,a] + IM0v[t-1,s,a]
            I.mv[t-1,a] <- I.mv[t-1,a] + IM0v[t-1,s,a] +
                                I0v1[t-1,s,a]*(1-(1-rVEsym1)*(1-rFm[a])) +
                                I0v2[t-1,s,a]*(1-(1-rVEsym2)*(1-rFm[a]))        # Among asym/mild: vaccinated and
                                                                                # after immunity waning in the recovered
            I.sq[t-1,a] <-I.sq[t-1,a] +QS[t-1,s,a]
           I.sqv[t-1,a] <-I.sqv[t-1,a] +QSv[t-1,s,a]
            I.s0[t-1,a] <-I.s0[t-1,a] +IS0[t-1,s,a]+ISq[t-1,s,a]+ISh[t-1,s,a]
             I.s[t-1,a] <- I.s[t-1,a] +IS0[t-1,s,a]+ISq[t-1,s,a]+ISh[t-1,s,a] +I0[t-1,s,a]*(1-rFm[a])

           I.s0v[t-1,a] <-I.s0v[t-1,a] +IS0v[t-1,s,a]+ISqv[t-1,s,a]+IShv[t-1,s,a]
            I.sv[t-1,a] <- I.sv[t-1,a] +IS0v[t-1,s,a]+ISqv[t-1,s,a]+IShv[t-1,s,a] +
                                                I0v1[t-1,s,a] *(1-rVEsym1)*(1-rFm[a]) +
                                                I0v2[t-1,s,a] *(1-rVEsym2)*(1-rFm[a])
                                }
                                }

#=============================================================================
# Estimating average no. of daily transmission per infectious individuals for estimating R0/Rt
```

```
        for(a in 1:10)          {
        for(j in 1:10)          {
        #-------------------------------------------------------------------------
                Ii0 <-(I.i0[t-1,j]+I.i0v[t-1,j])
                        if(Ii0>0)  {
                rIi0 <-(I.i0[t-1,j]+I.i0v[t-1,j]*InfI2)/(I.i0[t-1,j]+I.i0v[t-1,j])
                                        } else { rIi0 <-1 }
        beta2.i0[t-1] <-beta2.i0[t-1] +(beta[t-1]*CMTa[a,j]*adjCNTa[t-1,a]*
                        N[t-1,3,j]/N[t-1,3,11]) *(rFm[a]*ASYinf +(1-rFm[a])) *rIi0
        #-------------------------------------------------------------------------
                Im0 <-(I.m0[t-1,j]+I.m0v[t-1,j])
                        if(Im0>0){
                rIm0 <-(I.m0[t-1,j]+I.m0v[t-1,j]*InfI2)/(I.m0[t-1,j]+I.m0v[t-1,j])
                                        } else { rIm0 <-1 }
        beta2.m0[t-1] <-beta2.m0[t-1] +(beta[t-1]*CMTa[a,j]*adjCNTa[t-1,a]*
                        N[t-1,3,j]/N[t-1,3,11]) *ASYinf          * rIm0
        #-------------------------------------------------------------------------
                Is0 <-(I.s0[t-1,j]+I.s0v[t-1,j])
                        if(Is0>0)  {
                rIs0 <-(I.s0[t-1,j]+I.s0v[t-1,j]*InfI2)/(I.s0[t-1,j]+I.s0v[t-1,j])
                                        } else { rIs0 <-1 }
        beta2.s0[t-1] <-beta2.s0[t-1] +(beta[t-1]*CMTa[a,j]*adjCNTa[t-1,a]*
                        N[t-1,3,j]/N[t-1,3,11]) * rIs0
        #-------------------------------------------------------------------------
                Isq <-(I.sq[t-1,j]+I.sqv[t-1,j])
                        if(Isq>0)  {
                rIsq <-(I.sq[t-1,j]+I.sqv[t-1,j]*InfI2)/(I.sq[t-1,j]+I.sqv[t-1,j])
                                        } else { rIsq <-1 }
        beta2.sq[t-1] <-beta2.sq[t-1] +(beta[t-1]*CMTh[a,j]*
                        N[t-1,3,j]/N[t-1,3,11]) * rIsq
                        }
                        }

  #-------------------------------------------------------------------------------------
  # transition rate from I0 to IM0, etc.
  #-------------------------------------------------------------------------------------
        dk[t-1,1] <-alp2.a*   rFm.al[t-1]                                       # To IM0
        dk[t-1,2] <-alp2.a*(1-rFm.al[t-1])*(1-rFhos.al[t-1])*(1-rFsq.al[t-1])   # To IS0
        dk[t-1,3] <-alp2.a*(1-rFm.al[t-1])*(1-rFhos.al[t-1])*   rFsq.al[t-1]    # To ISq
        dk[t-1,4] <-alp2.a*(1-rFm.al[t-1])*   rFhos.al[t-1]                     # To ISh

        for(k in 1:4) { dA[t-1] <-dA[t-1] +dk[t-1,k]  }

        dk[t-1,5] <-alp2.a*(1-(1-rVEsym1)*(1-rFm.al[t-1]))                                  # To IM0v
        dk[t-1,6] <-alp2.a*   (1-rVEsym1)*(1-rFm.al[t-1])*(1-rFhos.al[t-1])*(1-rFsq.al[t-1])   # To IS0v
        dk[t-1,7] <-alp2.a*   (1-rVEsym1)*(1-rFm.al[t-1])*(1-rFhos.al[t-1])*   rFsq.al[t-1]    # To ISqv
        dk[t-1,8] <-alp2.a*   (1-rVEsym1)*(1-rFm.al[t-1])*   rFhos.al[t-1]                     # To IShv

        for(k in 5:8) { dAv1[t-1] <-dAv1[t-1] +dk[t-1,k]  }

        dk[t-1,9] <-alp2.a*(1-(1-rVEsym2)*(1-rFm.al[t-1]))                                  # To IM0v
        dk[t-1,10] <-alp2.a*   (1-rVEsym2)*(1-rFm.al[t-1])*(1-rFhos.al[t-1])*(1-rFsq.al[t-1])   # To IS0v
        dk[t-1,11] <-alp2.a*   (1-rVEsym2)*(1-rFm.al[t-1])*(1-rFhos.al[t-1])*   rFsq.al[t-1]    # To ISqv
        dk[t-1,12] <-alp2.a*   (1-rVEsym2)*(1-rFm.al[t-1])*   rFhos.al[t-1]                     # To IShv

        for(k in 9:12) { dAv2[t-1] <-dAv2[t-1] +dk[t-1,k]  }

#=============================================================
#  Estimating R0 and effective Rt
#-------------------------------------------------------------

        Rt[t-1] <- ((SU[t-1,3,11]+VA0[t-1,3,11])/N[t-1,3,11]) *
                (beta2.i0[t-1]                  /(dA[t-1])                  +
                 beta2.m0[t-1]*dk[t-1,1]        /(dA[t-1]*gam.m0a[t-1])     +
                 beta2.s0[t-1]*dk[t-1,2]        /(dA[t-1]*gam.s0a[t-1])     +
                 beta2.s0[t-1]*dk[t-1,3]        /(dA[t-1]*gam.s1a[t-1])     +
                 beta2.sq[t-1]*dk[t-1,3]*gam.s1a[t-1] /(dA[t-1]*gam.s2a[t-1])      +
                 beta2.s0[t-1]*dk[t-1,4]        /(dA[t-1]*gam.h1a[t-1]))    +
                   (VAef1[t-1,3,11]/N[t-1,3,11])*(1-rVEinf1) *
                (beta2.i0[t-1]                  /(dAv1[t-1])                +
```

```
                beta2.m0[t-1]*dk[t-1,5]              /(dAv1[t-1]*gam.m0a[t-1]) +
                beta2.s0[t-1]*dk[t-1,6]             /(dAv1[t-1]*gam.s0a[t-1]) +
                beta2.s0[t-1]*dk[t-1,7]             /(dAv1[t-1]*gam.s1a[t-1]) +
                beta2.sq[t-1]*dk[t-1,7]*gam.s1a[t-1] /(dAv1[t-1]*gam.s2a[t-1]) +
                beta2.s0[t-1]*dk[t-1,8]             /(dAv1[t-1]*gam.h1a[t-1])) +
                  (VAef2[t-1,3,11]/N[t-1,3,11])*(1-rVEinf2) *
               (beta2.i0[t-1]                        /(dAv2[t-1])                  +
                beta2.m0[t-1]*dk[t-1,9]              /(dAv2[t-1]*gam.m0a[t-1]) +
                beta2.s0[t-1]*dk[t-1,10]            /(dAv2[t-1]*gam.s0a[t-1]) +
                beta2.s0[t-1]*dk[t-1,11]            /(dAv2[t-1]*gam.s1a[t-1]) +
                beta2.sq[t-1]*dk[t-1,11]*gam.s1a[t-1] /(dAv2[t-1]*gam.s2a[t-1]) +
                beta2.s0[t-1]*dk[t-1,12]            /(dAv2[t-1]*gam.h1a[t-1])) +
                  ((SU.vac[t-1,3,11] +SU.rec[t-1,3,11])/N[t-1,3,11]) *
               (beta2.i0[t-1]                        /(dAv2[t-1])                  +
                beta2.m0[t-1]*dk[t-1,9]              /(dAv2[t-1]*gam.m0a[t-1]) +
                beta2.s0[t-1]*dk[t-1,10]            /(dAv2[t-1]*gam.s0a[t-1]) +
                beta2.s0[t-1]*dk[t-1,11]            /(dAv2[t-1]*gam.s1a[t-1]) +
                beta2.sq[t-1]*dk[t-1,11]*gam.s1a[t-1] /(dAv2[t-1]*gam.s2a[t-1]) +
                beta2.s0[t-1]*dk[t-1,12]            /(dAv2[t-1]*gam.h1a[t-1]))

#============================================================================
#  Susceptable infected by contacting infectious individuals with any age-sex
#   not quarantined or household isolated in the whole population
#----------------------------------------------------------------------------

          for(s in 1:2)          {
          for(a in 1:10)         {
          for(j in 1:10)         {

 # Temporary variable:
          ConInf <-  CMTa[a,j]*adjCNTa[t-1,a] *
            (ASYinf *(I.m[t-1,j] +InfI2 *I.mv[t-1,j]) +I.s[t-1,j] +InfI2 *I.sv[t-1,j])/N[t-1,3,j] +
                CMTh[a,j]* (I.sq[t-1,j] +InfI2 *I.sqv[t-1,j])/N[t-1,3,j]

 # Exposed from susceptible
          New.expsu[t-1,s,a] <-New.expsu[t-1,s,a] + beta[t-1] *SU[t-1,s,a] *ConInf
                                 }             # end age j
           SU[t-1,s,a] <-  SU[t-1,s,a] -New.expsu[t-1,s,a]
                                 }
                                 }

#-------------------------------------------------------
          for(s in 1:2)          {
          for(v in 1:dv0)        {
          for(a in 1:10)         {
          for(j in 1:10)         {

 # Temporary variable:
          ConInf <-  CMTa[a,j]*adjCNTa[t-1,a] *
            (ASYinf *(I.m[t-1,j] +InfI2 *I.mv[t-1,j]) +I.s[t-1,j] +InfI2 *I.sv[t-1,j])/N[t-1,3,j] +
                CMTh[a,j]* (I.sq[t-1,j] +InfI2 *I.sqv[t-1,j])/N[t-1,3,j]

 # Exposed in early vaccinated (<14 days):
               Expv0.temp <- beta[t-1] *eVA0[t-1,v,s,a] *ConInf
          Expv0[t-1,v,s,a] <- Expv0[t-1,v,s,a] +Expv0.temp
           eVA0[t-1,v,s,a] <-  eVA0[t-1,v,s,a] -Expv0.temp
                                 }
          New.expv0[t-1,s,a] <-New.expv0[t-1,s,a] +Expv0[t-1,v,s,a]
                                 }
                                 }
                                 }


#-------------------------------------------------------
          for(s in 1:2)          {
          for(w in 1:tv0)        {
          for(a in 1:10)         {
          for(j in 1:10)         {

 # Temporary variable:
```

```
            ConInf <-  CMTa[a,j]*adjCNTa[t-1,a] *
              (ASYinf *(I.m[t-1,j] +InfI2 *I.mv[t-1,j]) +I.s[t-1,j] +InfI2 *I.sv[t-1,j])/N[t-1,3,j] +
                    CMTh[a,j]* (I.sq[t-1,j] +InfI2 *I.sqv[t-1,j])/N[t-1,3,j]

  # Exposed in vaccinated dose-1 (risk reduced by "1-rVEinf1"):
            Expv1.temp <- beta[t-1] *eVAef1[t-1,w,s,a] *ConInf *(1-rVEinf1)
            Expv1[t-1,w,s,a] <-Expv1[t-1,w,s,a] + Expv1.temp
             eVAef1[t-1,w,s,a] <- eVAef1[t-1,w,s,a] -Expv1.temp
                                  }
            New.expv1[t-1,s,a] <-New.expv1[t-1,s,a] +Expv1[t-1,w,s,a]
                                  }
                                  }
                                  }

  #-------------------------------------------------------
            for(s in 1:2)         {
               tv1<-(t-vac.tim[1]); if(tv1<2) { tv1<-2  }
            for(w in 1:tv1)       {
            for(a in 1:10)        {
            for(j in 1:10)        {

  # Temporary variable:
            ConInf <-  CMTa[a,j]*adjCNTa[t-1,a] *
              (ASYinf *(I.m[t-1,j] +InfI2 *I.mv[t-1,j]) +I.s[t-1,j] +InfI2 *I.sv[t-1,j])/N[t-1,3,j] +
                    CMTh[a,j]* (I.sq[t-1,j] +InfI2 *I.sqv[t-1,j])/N[t-1,3,j]

  # Exposed in vaccinated dose-2 (risk reduced by "1-rVEinf2"):
            Expv2.temp <- beta[t-1] *eVAef2[t-1,w,s,a] *ConInf *(1-rVEinf2)
            Expv2[t-1,w,s,a] <-Expv2[t-1,w,s,a] + Expv2.temp
             eVAef2[t-1,w,s,a] <- eVAef2[t-1,w,s,a] -Expv2.temp
                                  }
            New.expv2[t-1,s,a] <-New.expv2[t-1,s,a] +Expv2[t-1,w,s,a]
                                  }
                                  }
                                  }

   #-------------------------------------------------------

            for(s in 1:2)         {
            for(a in 1:10)        {
            for(j in 1:10)        {

  # Temporary variable:
            ConInf <-  CMTa[a,j]*adjCNTa[t-1,a] *
              (ASYinf *(I.m[t-1,j] +InfI2 *I.mv[t-1,j]) +I.s[t-1,j] +InfI2 *I.sv[t-1,j])/N[t-1,3,j] +
                    CMTh[a,j]* (I.sq[t-1,j] +InfI2 *I.sqv[t-1,j])/N[t-1,3,j]

  # Exposed due to loss of immunity in vaccinated/recovered (risk fraction: "rLOSinf"):
             New.explosv1[t-1,s,a]  <- New.explosv1[t-1,s,a] + beta[t-1] *(SU.vac[t-1,s,a]) *ConInf
             New.explosrec[t-1,s,a] <- New.explosrec[t-1,s,a] + beta[t-1] *(SU.rec[t-1,s,a]) *ConInf
                                  }

             SU.vac[t-1,s,a] <-SU.vac[t-1,s,a] - New.explosv1[t-1,s,a]
             SU.rec[t-1,s,a] <-SU.rec[t-1,s,a] -New.explosrec[t-1,s,a]
                                  }
                                  }


#########################################################

            if(VAC==1)                    {           ### Vaccination yes=1 or no=0

#=====================================================================
#  Total vaccinated and days required to complete vaccination
            if(t <vac.tim[1]+1 ) { Vtru<-0; Vp<-7; Rv<-0 }  # Vtru: vaccination true-1 or false-0
              for(rev in 1:nRvac)                 {           # from 1 to nRvac -no. of vac programs
            if(t== (vac.tim[rev]+1) )       {
                                  Vtru <-1;   Vp <-1;   Rv <-rev
                        T.vac[Rv,1] <- vac.tim[rev] +1          # T.vac starting time for age Vp==1
                                                }
```

```
                                                }       # end for rev in 1:nRvac
#=============================================================================

        if( Vtru==1)        {

  #-------------------------------------------------------------
        for(nVp in 2:6)             {
         if(t==T.vac[Rv,nVp]) {  Vp <-nVp  }
                                    }
  #-------------------------------------------------------------
                Totalvac <-0
  #-----------------------------------------------------------------------------

           vaccrg <-vac.crg4                    # if Vp<4: age 50+

        if(Vp==6)           {
        if(Rv==1) { vaccrg <-vac.crg1*0.4 }  # 16-19  in 10-19
        if(Rv >1) { vaccrg <-vac.crg1*0.1 }  # new 16 in 10-19
                            }       # end if Vp=6

        if(Vp==5)           {
            vaccrg <-vac.crg2   # age 20-39
                            }       # end if Vp=5

        if(Vp==4)           {
            vaccrg <-vac.crg3   # age 40-49
                            }       # end if Vp=4

        #-----------------------------------------------------------------------------

        for(a in vagp[Vp,1]:vagp[Vp,2])     {       # a from low to high age limit
        for(s in 1:2)                       {       # s in 1:2

  # Vaccinated among SU, RE, and lost immunity:
  # In not-exposed only

        Totalvac <-Totalvac + (SU[t-1,s,a] +    RE[t-1,s,a] +SU.vac[t-1,s,a] +SU.rec[t-1,s,a] ) *vaccrg


  # vaccination of people vaccinated for at leas 91 days (3 months)

        vcrg <- vaccrg
        if(Vp==6) { vcrg <- vac.crg1 }
        if(Rv>1) {
                for(w in 91:t)      {
           Totalvac <- Totalvac + eVAef2[t-1,w,s,a] *vcrg
                                    }
                                    }  # end Rv>1
                                        }       # end s
                                        }       # end a

  #-----------------------------------------------------------------------------
        if( t==T.vac[Rv,Vp] )               {       # Fixing V.day, V.dmax, vacend estimates at T.vac[Rv,Vp]
         V.day[Rv,Vp] <- round(Totalvac/vac.dmaxi) +1  # V.day: no. of days required for Vp phase
        V.dmax[Rv,Vp] <-Totalvac/V.day[Rv,Vp]               # Totalvac at the 1st day of a vac phase
        vacend[Rv,Vp] <- T.vac[Rv,Vp] +V.day[Rv,Vp] -1
        if(Vp<6)  {  T.vac[Rv,Vp+1] <-vacend[Rv,Vp] +1  }
                                        }       # end if t==T.vac
   #----------------------------------------------------------------------------------

                                    }       # end if Vtru==1
                                    }       # end if VAC==1

#=============================================================
# Estimating VA, SU, EX, I, and RE, hospitalised, ICU, and no. of tests required

  for(s in 1:2)     {       ### sex male-1, female-2
  for(a in 1:10)    {       ### age group 1-10

        if(VAC==1)              {       # Vaccination: 1-yes or 0-no
```

```
            if(Vtru==1)                     {
            if(Vp<(vac.AGP[Rv]+1) )     {           # only apply to Vp 1-6
#-------------------------------------------------------------------------------
               vaccrg <-vac.crg4                    # if Vp<4: age 50+

            if(Vp==6)           {
            if(Rv==1) { vaccrg <-vac.crg1*0.4 }  # 18-19 in 10-19
            if(Rv >1) { vaccrg <-vac.crg1*0.1 }  # new 16 in 10-19
                                }           # end if Vp=6
            if(Vp==5)           {
                    vaccrg <-vac.crg2   # age 20-39
                                }           # end if Vp=5
            if(Vp==4)           {
                    vaccrg <-vac.crg3   # age 40-49
                                }           # end if Vp=4
#-------------------------------------------------------------------------------
            if(a>vagp[Vp,1]-1) {
            if(a<vagp[Vp,2]+1){

  # Vaccinating susceptible
 #-------------------------------------------------------------------------------
               vac.SU[t-1,s,a] <- V.dmax[Rv,Vp] *(SU[t-1,s,a]*vaccrg)/Totalvac
            if(vac.SU[t-1,s,a]>SU[t-1,s,a]) {   vac.SU[t-1,s,a] <-SU[t-1,s,a]*vaccrg }
                        SU[t-1,s,a] <-SU[t-1,s,a] -vac.SU[t-1,s,a]
               New.vac[t-1,s,a] <- vac.SU[t-1,s,a]

    # Vaccinating loss of immunity in vaccinated
    #-------------------------------------------------------------------------------
               vac.SUvac[t-1,s,a] <- V.dmax[Rv,Vp] *(SU.vac[t-1,s,a]*vaccrg)/Totalvac
            if(vac.SUvac[t-1,s,a]>SU.vac[t-1,s,a]) { vac.SUvac[t-1,s,a] <-SU.vac[t-1,s,a]*vaccrg }
                        SU.vac[t-1,s,a] <- SU.vac[t-1,s,a] -vac.SUvac[t-1,s,a]

               New.vac[t-1,s,a] <-New.vac[t-1,s,a] +vac.SUvac[t-1,s,a]

    # Vaccinating loss of immunity in recovered
    #-------------------------------------------------------------------------------
               vac.SUrec[t-1,s,a] <- V.dmax[Rv,Vp] *(SU.rec[t-1,s,a]*vaccrg)/Totalvac
            if(vac.SUrec[t-1,s,a]>SU.rec[t-1,s,a]) { vac.SUrec[t-1,s,a] <-SU.rec[t-1,s,a]*vaccrg }
                        SU.rec[t-1,s,a] <- SU.rec[t-1,s,a] -vac.SUrec[t-1,s,a]
               New.vac[t-1,s,a] <-New.vac[t-1,s,a] +vac.SUrec[t-1,s,a]

    # Vaccinating all recovered
                        vac.RE<-0
               for(w in 1:t)        {
                        vac.RE <-V.dmax[Rv,Vp] *(eRE[t-1,w,s,a]*vaccrg)/Totalvac
               if(vac.RE>eRE[t-1,w,s,a]) { vac.RE <-eRE[t-1,w,s,a]*vaccrg  }
                        eRE[t-1,w,s,a] <- eRE[t-1,w,s,a] -vac.RE
                        eRE[t,1,s,a] <- eRE[t,1,s,a] +vac.RE *(1-rADth0[t.dr,s,a])   # Boost RE status after vaccination
             New.vac[t-1,s,a] <- New.vac[t-1,s,a] +vac.RE
                                        } # end w in 1:t

    # Re-vaccinating previous vaccinated, if Rv>1
    #---------------------------------------------------------------
            if(Rv>1) {
            vcrg <- vaccrg
            if(Vp==6) { vcrg <- vac.crg1 }          # for age 16-19
                        vac.VAef <-0
               for(w in 91:t)       {
                        vac.VAef <-V.dmax[Rv,Vp] *(eVAef2[t-1,w,s,a]*vcrg)/Totalvac
               if(vac.VAef>eVAef2[t-1,w,s,a]) { vac.VAef <-eVAef2[t-1,w,s,a]*vcrg  }
                        eVAef2[t-1,w,s,a] <- eVAef2[t-1,w,s,a] -vac.VAef
                          eVAef2[t,1,s,a] <- eVAef2[t,1,s,a] +vac.VAef *(1-rADth0[t.dr,s,a])
             New.vac[t-1,s,a] <- New.vac[t-1,s,a] +vac.VAef
                                        } # end w in 91:t
                        }  # End if Rv>1

    #---------------------------------------------------------------
                                }           # end a>a1-1
                                }           # end a<a2+1
```

```
          if( Vp==vac.AGP[Rv] )        {
             if(t==vacend[Rv,Vp]+1)  {  Vtru <-0  }
                                        }
                                        }          # end if Vp<7
                                        }          # end if Vtru==1

#=========================================================================
#  Calculating no. of VA status at time t

          eVA0[t,1,s,a] <- vac.SU[t-1,s,a] *(1-rADth0[t.dr,s,a])
          VA0[t,s,a] <-eVA0[t,1,s,a]
          for(v in 2:14)         {
           eVA0[t,v,s,a] <-eVA0[t-1,v-1,s,a] *(1-rADth0[t.dr,s,a])
             VA0[t,s,a] <-VA0[t,s,a] +eVA0[t,v,s,a]
                                }          # end for v in 2:14

          #-------------------------------------------------------------
           eVAef1[t,1,s,a] <-eVA0[t-1,14,s,a]*(1-rADth0[t.dr,s,a])
          VAef1[t,s,a] <- eVAef1[t,1,s,a]
          for(w in 2:63)         {
           eVAef1[t,w,s,a] <-eVAef1[t-1,w-1,s,a]*(1-rADth0[t.dr,s,a])
             VAef1[t,s,a] <-VAef1[t,s,a] +eVAef1[t,w,s,a]
                                }

          #-------------------------------------------------------------
          #  eVAef2 include renewed vaccination of previously vaccinated
          #-------------------------------------------------------------
           eVAef2[t,1,s,a] <- eVAef2[t,1,s,a] +(eVAef1[t-1,63,s,a] +
                              vac.SUvac[t-1,s,a] +vac.SUrec[t-1,s,a])*(1-rADth0[t.dr,s,a])
                    tv1<-(t-vac.tim[1]); if(tv1<3) { tv1<-3  }
          for(w in 2:tv1)        {
           eVAef2[t,w,s,a] <- eVAef2[t-1,w-1,s,a] *(1-gimmv.wan[w-1])*(1-rADth0[t.dr,s,a])
                                }
          for(w in 1:tv1)        {
                    VAef2[t,s,a] <-VAef2[t,s,a] +eVAef2[t,w,s,a]
                                }

                                }   ###  end if VAC==1

 #=============================================================================
 # Estimating return to susceptible in the recovered/vaccinated, minus newly infected due to loss of immunity
 #-----------------------------------------------------------------------------

                    SUvac <-0
             tv1 <-(t-vac.tim[1]); if(tv1<2) { tv1<-2  }        # Loss of vaccine immunity
           for(w in 1:tv1)       {
            SUvac <- SUvac +eVAef2[t-1,w,s,a] *gimmv.wan[w]
                                        }
            SU.vac[t,s,a] <-(SU.vac[t-1,s,a] +SUvac) *(1-rADth0[t.dr,s,a])

                    SUrec  <-0
             tw1 <-(t-ts); if(tw1<2) { tw1<-2  }                # Loss of natural immunity
           for(w in 1:tw1)       {
            SUrec <- SUrec  +eRE[t-1,w,s,a] *gimm.wan[w]
                                        }
            SU.rec[t,s,a]<- (SU.rec[t-1,s,a] +SUrec) *(1-rADth0[t.dr,s,a])

#=============================================================================
# Estimating SU, EX, I, and RE
#   New exposed and vaccinated already removed from SU[t-1, ...]
#=============================================================================
          SU[t,s,a] <-(SU[t-1,s,a])*(1-rADth0[t.dr,s,a]) +
                                              dBth[a]*Bsex[s] -Seed.all[t,s,a]

           New.expal[t-1,s,a] <-  New.expsu[t-1,s,a]    +New.expv0[t-1,s,a] +
                                  New.expv1[t-1,s,a]    +New.expv2[t-1,s,a] +
                                  New.explosv1[t-1,s,a] +New.explosrec[t-1,s,a]

 #=============================================================================
 # All newsymptomatic cases at t. rVEsym: efficacy of vaccine on symptomatic cases
```

```
#-------------------------------------------------------------------------
        for(d in 2:dd)          {
    New.sym[t,s,a] <- New.sym[t,s,a] + (eI0[t-1,d-1,s,a] +
                                            eI0v1[t-1,d-1,s,a]*(1-rVEsym1) +
                                            eI0v2[t-1,d-1,s,a]*(1-rVEsym2) ) *alp2[d-1]*(1-rFm[a])
                            }   # end for d in 2:dd

         eEX[t,1,s,a] <-(New.expsu[t-1,s,a] +New.expv0[t-1,s,a]) +eSeed[t,1,s,a]

        eEXv1[t,1,s,a] <-New.expv1[t-1,s,a]

        eEXv2[t,1,s,a] <-(New.expv2[t-1,s,a] +New.explosv1[t-1,s,a] +New.explosrec[t-1,s,a])

        # All covid deaths from symptomatic cases, ie, (1-rFm)
        #-------------------------------------------------------------------------------------
          eInf[t,1,s,a] <- eEX[t,1,s,a] *(1-rFm[a]) *drisk.inf[s,a]*adjDth[t]
          eInfv[t,1,s,a] <- (eEXv1[t,1,s,a] *(1-rVEsym1)*(1-rFm[a]) +
                                  eEXv2[t,1,s,a] *(1-rVEsym2)*(1-rFm[a]) ) *drisk.inf[s,a]*adjDth[t]
        #-------------------------------------------------------------------------------------
                    I0.1   <-0; I0v1.1 <-0; I0v2.1 <-0

        for(d in 2:dd)          {   # d in 2:dd=60
         I0.1  <- I0.1   +  eI0[t-1,d-1,s,a]*alp2[d-1]
         I0v1.1 <-I0v1.1 +eI0v1[t-1,d-1,s,a]*alp2[d-1]
         I0v2.1 <-I0v2.1 +eI0v2[t-1,d-1,s,a]*alp2[d-1]

          eI0[t,1,s,a] <-   eI0[t,1,s,a] +   eEX[t-1,d-1,s,a]*alp1[d-1]              # From Exp to I0
         eI0v1[t,1,s,a] <-eI0v1[t,1,s,a] + eEXv1[t-1,d-1,s,a]*alp1[d-1]          # Exposed in vaccinated dose-1
         eI0v2[t,1,s,a] <-eI0v2[t,1,s,a] + eEXv2[t-1,d-1,s,a]*alp1[d-1]          # Exposed in vaccinated dose-2

          eQS[t,1,s,a] <- eQS[t,1,s,a] + eISq[t-1,d-1,s,a]*gam.s1[t-1,d-1]
         eQSv[t,1,s,a] <-eQSv[t,1,s,a] +eISqv[t-1,d-1,s,a]*gam.s1[t-1,d-1]
          eHS[t,1,s,a] <- eHS[t,1,s,a] + eISh[t-1,d-1,s,a]*gam.h1[t-1,d-1]
         eHSv[t,1,s,a] <-eHSv[t,1,s,a] +eIShv[t-1,d-1,s,a]*gam.h1[t-1,d-1]

                            }   # end d in 2:dd

        #-------------------------------------------------------------------------------------------
        # Allocate I0.all to IM, IS, etc
        #-------------------------------------------------------------------------------------------
         eIM0[t,1,s,a] <- I0.1 *rFm[a]
         eIS0[t,1,s,a] <- I0.1 *(1-rFm[a]) *(1-rFhos[t-1,a])*(1-rFsq[t-1,a])
         eISq[t,1,s,a] <- I0.1 *(1-rFm[a]) *(1-rFhos[t-1,a])*   rFsq[t-1,a]
         eISh[t,1,s,a] <- I0.1 *(1-rFm[a]) *   rFhos[t-1,a]

         eIM0v[t,1,s,a] <- I0v1.1* (1-(1-rVEsym1)*(1-rFm[a])) +I0v2.1*(1-(1-rVEsym2)*(1-rFm[a]))
         eIS0v[t,1,s,a] <-(I0v1.1* (1-rVEsym1)+I0v2.1*(1-rVEsym2))*(1-rFm[a])*(1-rFhos[t-1,a])*(1-rFsq[t-1,a])
         eISqv[t,1,s,a] <-(I0v1.1* (1-rVEsym1)+I0v2.1*(1-rVEsym2))*(1-rFm[a])*(1-rFhos[t-1,a])*   rFsq[t-1,a]
         eIShv[t,1,s,a] <-(I0v1.1* (1-rVEsym1)+I0v2.1*(1-rVEsym2))*(1-rFm[a])*   rFhos[t-1,a]

        #=========================================================================
        # Estimate covid related deaths by days since exposed/infected
        # ------------------------------------------------------------------
        for(d in 2:dd+1)      {
           nDth.inf[t-1,s,a] <- nDth.inf[t-1,s,a]+ eInf[t-1,d-1,s,a]*gdth.inf[d-1]
          nDth.infv[t-1,s,a] <-nDth.infv[t-1,s,a]+eInfv[t-1,d-1,s,a]*gdth.inf[d-1]
                                    }

        #=========================================================================
        # UK data: 64% covid-19 deaths in hospital, 36% in other settings
        if(HS[t-1,s,a]>0) { r.dhos<-nDth.inf[t-1,s,a] *0.64/HS[t-1,s,a] } else { r.dhos<-0 }
        if(r.dhos>0.9)          { r.dhos<-0.9 }

        for(d in 1:dd)                  {
          nDth.hos[t-1,d,s,a] <- eHS[t-1,d,s,a]*r.dhos
                    if(nDth.hos[t-1,d,s,a]<0) { nDth.hos[t-1,d,s,a]<-0  }
                                    }

        for(d in 1:dd)          {                       # All hospital deaths
         nDthHos.all[t-1,s,a] <-nDthHos.all[t-1,s,a] +nDth.hos[t-1,d,s,a]
```

```
                                }

        nDth.cov[t-1,s,a] <-nDth.cov[t-1,s,a] +nDthHos.all[t-1,s,a]  # sum up covid deaths

  # Deaths outside hospitals in IS0 and QS:
        nDth.ss[t-1,s,a] <- nDth.inf[t-1,s,a] -nDthHos.all[t-1,s,a]
                if(nDth.ss[t-1,s,a]<0) { nDth.ss[t-1,s,a]<-0  }

           nS0Sq <- IS0[t-1,s,a] +QS[t-1,s,a]                  # Temporary variable

        if(nS0Sq>0) { r.dss<-nDth.ss[t-1,s,a]/nS0Sq } else { r.dss<-0              }
        if(r.dss>1) { r.dss<-0.99 }

         for(d in 1:dd)                         {
          nDth.s0[t-1,d,s,a] <-eIS0[t-1,d,s,a] *r.dss
                if(nDth.s0[t-1,d,s,a]<0) { nDth.s0[t-1,d,s,a]<-0  }
          nDth.sq[t-1,d,s,a] <- eQS[t-1,d,s,a] *r.dss
                if(nDth.sq[t-1,d,s,a]<0) { nDth.sq[t-1,d,s,a]<-0  }

        nDth.cov[t-1,s,a] <-nDth.cov[t-1,s,a] +nDth.s0[t-1,d,s,a] +nDth.sq[t-1,d,s,a]      ## sum up covid deaths

                                        }

        #-----------------------------------------------------
        # Repeat the above for covid deaths in reinfected
        #-----------------------------------------------------

        if(HSv[t-1,s,a]>0) { r.dhosv<-nDth.infv[t-1,s,a] *0.64/HSv[t-1,s,a] } else { r.dhosv<-0 }
        if(r.dhosv>0.9)       { r.dhosv<-0.9 }

        for(d in 1:dd)                          {
          nDth.hosv[t-1,d,s,a] <- eHSv[t-1,d,s,a]*r.dhosv
                if(nDth.hosv[t-1,d,s,a]<0) { nDth.hosv[t-1,d,s,a]<-0  }
                                        }

        for(d in 1:dd)          {
         nDthHos.allv[t-1,s,a] <-nDthHos.allv[t-1,s,a] +nDth.hosv[t-1,d,s,a]
                                }

        nDth.cov[t-1,s,a] <-nDth.cov[t-1,s,a] +nDthHos.allv[t-1,s,a]              # sum up covid deaths

  # Deaths outside hospitals in IS0 and QS:
        nDth.ssv[t-1,s,a] <- nDth.infv[t-1,s,a] -nDthHos.allv[t-1,s,a]
                if(nDth.ssv[t-1,s,a]<0) { nDth.ssv[t-1,s,a]<-0  }

           nS0Sqv <- IS0v[t-1,s,a] +QSv[t-1,s,a]                        # Temporary variable

        if(nS0Sqv>0) { r.dssv<-nDth.ssv[t-1,s,a]/nS0Sqv } else { r.dssv<-0    }
        if(r.dssv>1) { r.dssv<-0.99 }

         for(d in 1:dd)                         {
          nDth.s0v[t-1,d,s,a] <-eIS0v[t-1,d,s,a] *r.dssv
                if(nDth.s0v[t-1,d,s,a]<0) { nDth.s0v[t-1,d,s,a]<-0  }
          nDth.sqv[t-1,d,s,a] <- eQSv[t-1,d,s,a] *r.dssv
                if(nDth.sqv[t-1,d,s,a]<0) { nDth.sqv[t-1,d,s,a]<-0  }

        nDth.cov[t-1,s,a] <-nDth.cov[t-1,s,a] +nDth.s0v[t-1,d,s,a] +nDth.sqv[t-1,d,s,a]   ## sum up covid deaths

                                        }

#=============================================================================================-

        for(d in 2:dd)          {

          eEX[t,d,s,a] <-  eEX[t-1,d-1,s,a]*(1-alp1[d-1]) +eSeed[t,d,s,a]
         eEXv1[t,d,s,a] <-eEXv1[t-1,d-1,s,a]*(1-alp1[d-1])
         eEXv2[t,d,s,a] <-eEXv2[t-1,d-1,s,a]*(1-alp1[d-1])

          eI0[t,d,s,a] <-  eI0[t-1,d-1,s,a]*(1-alp2[d-1])
         eI0v1[t,d,s,a] <-eI0v1[t-1,d-1,s,a]*(1-alp2[d-1])
```

```
 eI0v2[t,d,s,a] <-eI0v2[t-1,d-1,s,a]*(1-alp2[d-1])

eIM0[t,d,s,a] <- eIM0[t-1,d-1,s,a]*(1-gam.m0[t-1,d-1])
eISq[t,d,s,a] <- eISq[t-1,d-1,s,a]*(1-gam.s1[t-1,d-1])
eISh[t,d,s,a] <- eISh[t-1,d-1,s,a]*(1-gam.h1[t-1,d-1])

eIS0[t,d,s,a] <-(eIS0[t-1,d-1,s,a]- nDth.s0[t-1,d-1,s,a])*(1-gam.s0[t-1,d-1])
 eQS[t,d,s,a] <- (eQS[t-1,d-1,s,a]- nDth.sq[t-1,d-1,s,a])*(1-gam.s2[t-1,d-1])
 eHS[t,d,s,a] <- (eHS[t-1,d-1,s,a]-nDth.hos[t-1,d-1,s,a])*(1-gam.h2[t-1,d-1])

eIM0v[t,d,s,a] <- eIM0v[t-1,d-1,s,a]*(1-gam.m0[t-1,d-1])
eISqv[t,d,s,a] <- eISqv[t-1,d-1,s,a]*(1-gam.s1[t-1,d-1])
eIShv[t,d,s,a] <- eIShv[t-1,d-1,s,a]*(1-gam.h1[t-1,d-1])

eIS0v[t,d,s,a] <-(eIS0v[t-1,d-1,s,a]- nDth.s0v[t-1,d-1,s,a])*(1-gam.s0[t-1,d-1])
 eQSv[t,d,s,a] <- (eQSv[t-1,d-1,s,a]- nDth.sqv[t-1,d-1,s,a])*(1-gam.s2[t-1,d-1])
 eHSv[t,d,s,a] <- (eHSv[t-1,d-1,s,a]-nDth.hosv[t-1,d-1,s,a])*(1-gam.h2[t-1,d-1])

eInf[t,d,s,a] <- eInf[t-1,d-1,s,a]*(1-gdth.inf[d-1])

eInfv[t,d,s,a] <- eInfv[t-1,d-1,s,a]*(1-gdth.inf[d-1])

                                    } # End d in 2:dd

#=============================================================================
#  Overall EX, I0, IM, IS, etc
#=============================================================================
          for(d in 1:dd)          {   # d in 1:dd

 EX[t,s,a] <-  EX[t,s,a] + eEX[t,d,s,a]
 I0[t,s,a] <-  I0[t,s,a] + eI0[t,d,s,a]
IM0[t,s,a] <- IM0[t,s,a] +eIM0[t,d,s,a]
IS0[t,s,a] <- IS0[t,s,a] +eIS0[t,d,s,a]
ISq[t,s,a] <- ISq[t,s,a] +eISq[t,d,s,a]
 QS[t,s,a] <-  QS[t,s,a] + eQS[t,d,s,a]
ISh[t,s,a] <- ISh[t,s,a] +eISh[t,d,s,a]
 HS[t,s,a] <-  HS[t,s,a] + eHS[t,d,s,a]

EXv1[t,s,a] <- EXv1[t,s,a] +eEXv1[t,d,s,a]
I0v1[t,s,a] <- I0v1[t,s,a] +eI0v1[t,d,s,a]

EXv2[t,s,a] <- EXv2[t,s,a] +eEXv2[t,d,s,a]
I0v2[t,s,a] <- I0v2[t,s,a] +eI0v2[t,d,s,a]

IM0v[t,s,a] <- IM0v[t,s,a] +eIM0v[t,d,s,a]
IS0v[t,s,a] <- IS0v[t,s,a] +eIS0v[t,d,s,a]
ISqv[t,s,a] <- ISqv[t,s,a] +eISqv[t,d,s,a]
 QSv[t,s,a] <-  QSv[t,s,a] + eQSv[t,d,s,a]
IShv[t,s,a] <- IShv[t,s,a] +eIShv[t,d,s,a]
 HSv[t,s,a] <-  HSv[t,s,a] + eHSv[t,d,s,a]

                          }   # end for d in 1:dd

 HS.new[t,s,a] <-eHS[t,1,s,a]  +eHSv[t,1,s,a]       # New hospital admission

#=============================================================================
#  Recovered from infected
#-----------------------------------------------------
          for(d in 1:dd)                  {
   eRE[t,1,s,a] <-eRE[t,1,s,a] +
                (eQS[t-1,d,s,a]-  nDth.sq[t-1,d,s,a])*gam.s2[t-1,d] +
          (eIS0[t-1,d,s,a]-  nDth.s0[t-1,d,s,a])*gam.s0[t-1,d] +
                (eHS[t-1,d,s,a]- nDth.hos[t-1,d,s,a])*gam.h2[t-1,d] +
                eIM0[t-1,d,s,a]                  *gam.m0[t-1,d] +
                (eQSv[t-1,d,s,a]-  nDth.sqv[t-1,d,s,a])*gam.s2[t-1,d] +
          (eIS0v[t-1,d,s,a]-  nDth.s0v[t-1,d,s,a])*gam.s0[t-1,d] +
                (eHSv[t-1,d,s,a]- nDth.hosv[t-1,d,s,a])*gam.h2[t-1,d] +
                eIM0v[t-1,d,s,a]                  *gam.m0[t-1,d]

                                    }  # end for d in 1:dd
```

```
            RE[t,s,a] <-eRE[t,1,s,a]

                        tw1 <-(t-ts); if(tw1<3) { tw1<-3  }
             for(w in 2:tw1)      {
              eRE[t,w,s,a] <- eRE[t-1,w-1,s,a]*(1-rADth0[t.dr,s,a]) *(1-gimm.wan[w-1])

                 if(w==tw1)   {
              eRE[t,w,s,a] <-(eRE[t,w-1,s,a]+eRE[t-1,w,s,a]) *(1-rADth0[t.dr,s,a]) *(1-gimm.wan[tw1])
                                  }

            RE[t,s,a] <-RE[t,s,a] +eRE[t,w,s,a]

                                          }            # end w in 2:tw1

#---------------------------------------------------------------------------
             N[t,s,a] <-    SU[t,s,a] + SU.vac[t,s,a] + SU.rec[t,s,a] +
                                  EX[t,s,a] +   EXv1[t,s,a] + EXv2[t,s,a] +
                                  I0[t,s,a] +   I0v1[t,s,a] + I0v2[t,s,a] +
                                 IM0[t,s,a] +          IS0[t,s,a] +
                                 ISq[t,s,a] +    QS[t,s,a] +
                                 ISh[t,s,a] +           HS[t,s,a] +
                                IM0v[t,s,a] +  IS0v[t,s,a] +
                                ISqv[t,s,a] +   QSv[t,s,a] +
                                IShv[t,s,a] +          HSv[t,s,a] +
                              VA0[t,s,a] +  VAef1[t,s,a] + VAef2[t,s,a] +
                                 RE[t,s,a]

#---------------------------------------------------------------------------
                                } #---------End for age in 1:18    ######
                                } #---------End for sex in 1:2     ######

                    }    ###### end if t>ts ########

#============================================================================
#  Upgrading age at the beginning of a year (t1)
#    jN[a], etc, are used to record N[a], etc, before their adjustments, because
#     unchanged N[a] is required for the adjustment of N[a+1], etc.
#----------------------------------------------------------------------------
            if(t==tage)             {
            for(s in 1:2)           {
            for(a in 1:10)          {
            if(a>1)   { Ag1<-1; Ag2<-1; Ag3<-10; Ag4<-10; a1<-(a-1) }          # for age 10 to 79
            if(a==1)  { Ag1<-1; Ag2<-0; Ag3<-5;  Ag4<-5;  a1<-1     }          # for age 0-4
            if(a==2)  { Ag1<-1; Ag2<-1; Ag3<-5;  Ag4<-5;  a1<-1     }          # for age 5-9
            if(a==3)  { Ag1<-1; Ag2<-1; Ag3<-10; Ag4<-5;  a1<-2     }          # for age 10-19
            if(a==10) { Ag1<-0; Ag2<-1; Ag3<-10; Ag4<-10; a1<-9     }          # for age 80+
                         jN[a] <- N[t,s,a]
                         N[t,s,a] <-   N[t,s,a] - Ag1*  N[t,s,a]/(Ag3)  + Ag2* jN[a1]/(Ag4)
                         jSU[a] <- SU[t,s,a]
                        SU[t,s,a] <-  SU[t,s,a] - Ag1* SU[t,s,a]/(Ag3)  +Ag2* jSU[a1]/(Ag4)
                         jRE[a] <- RE[t,s,a]
                        RE[t,s,a] <-  RE[t,s,a] - Ag1* RE[t,s,a]/(Ag3)  +Ag2* jRE[a1]/(Ag4)
                         jEX[a] <- EX[t,s,a]
                        EX[t,s,a] <-  EX[t,s,a] - Ag1* EX[t,s,a]/(Ag3)  +Ag2* jEX[a1]/(Ag4)
                         jI0[a] <- I0[t,s,a]
                        I0[t,s,a] <-  I0[t,s,a] - Ag1* I0[t,s,a]/(Ag3)  +Ag2* jI0[a1]/(Ag4)
                         jIM0[a] <- IM0[t,s,a]
                       IM0[t,s,a] <- IM0[t,s,a] -Ag1* IM0[t,s,a]/(Ag3)  +Ag2*jIM0[a1]/(Ag4)
                         jIS0[a] <- IS0[t,s,a]
                       IS0[t,s,a] <- IS0[t,s,a] -Ag1* IS0[t,s,a]/(Ag3)  +Ag2*jIS0[a1]/(Ag4)
                         jISq[a] <- ISq[t,s,a]
                       ISq[t,s,a] <- ISq[t,s,a] -Ag1* ISq[t,s,a]/(Ag3)  +Ag2*jISq[a1]/(Ag4)
                         jISh[a] <- ISh[t,s,a]
                       ISh[t,s,a] <- ISh[t,s,a] -Ag1* ISh[t,s,a]/(Ag3)  +Ag2*jISh[a1]/(Ag4)
                         jQS[a] <- QS[t,s,a]
                        QS[t,s,a] <-  QS[t,s,a] -Ag1*  QS[t,s,a]/(Ag3)  +Ag2* jQS[a1]/(Ag4)
                         jHS[a] <- HS[t,s,a]
                        HS[t,s,a] <-  HS[t,s,a] -Ag1*  HS[t,s,a]/(Ag3)  +Ag2* jHS[a1]/(Ag4)
```

```
  jEXv1[a] <- EXv1[t,s,a]
EXv1[t,s,a] <- EXv1[t,s,a] -  Ag1*  EXv1[t,s,a]/(Ag3)  +Ag2* jEXv1[a1]/(Ag4)
  jI0v1[a] <- I0v1[t,s,a]
I0v1[t,s,a] <- I0v1[t,s,a] -  Ag1*  I0v1[t,s,a]/(Ag3)  +Ag2* jI0v1[a1]/(Ag4)
  jEXv2[a] <- EXv2[t,s,a]
EXv2[t,s,a] <- EXv2[t,s,a] -  Ag1*  EXv2[t,s,a]/(Ag3)  +Ag2* jEXv2[a1]/(Ag4)
  jI0v2[a] <- I0v2[t,s,a]
I0v2[t,s,a] <- I0v2[t,s,a] -  Ag1*  I0v2[t,s,a]/(Ag3)  +Ag2* jI0v2[a1]/(Ag4)

  jIM0v[a] <- IM0v[t,s,a]
IM0v[t,s,a] <- IM0v[t,s,a] -Ag1* IM0v[t,s,a]/(Ag3)  +Ag2*jIM0v[a1]/(Ag4)
  jIS0v[a] <- IS0v[t,s,a]
IS0v[t,s,a] <- IS0v[t,s,a] -Ag1* IS0v[t,s,a]/(Ag3)  +Ag2*jIS0v[a1]/(Ag4)
  jISqv[a] <- ISq[t,s,a]
ISqv[t,s,a] <- ISqv[t,s,a] -Ag1* ISqv[t,s,a]/(Ag3)  +Ag2*jISqv[a1]/(Ag4)
  jIShv[a] <- IShv[t,s,a]
IShv[t,s,a] <- IShv[t,s,a] -Ag1* IShv[t,s,a]/(Ag3)  +Ag2*jIShv[a1]/(Ag4)
  jQSv[a] <- QSv[t,s,a]
 QSv[t,s,a] <-  QSv[t,s,a] -Ag1*  QSv[t,s,a]/(Ag3)  +Ag2* jQSv[a1]/(Ag4)
  jHSv[a] <- HSv[t,s,a]
 HSv[t,s,a] <-  HSv[t,s,a] -Ag1*  HSv[t,s,a]/(Ag3)  +Ag2* jHSv[a1]/(Ag4)
  jSU.vac[a] <- SU.vac[t,s,a]
 SU.vac[t,s,a] <-  SU.vac[t,s,a] -Ag1* SU.vac[t,s,a]/(Ag3)  +Ag2* jSU.vac[a1]/(Ag4)
  jSU.rec[a] <- SU.rec[t,s,a]
 SU.rec[t,s,a] <-  SU.rec[t,s,a] -Ag1* SU.rec[t,s,a]/(Ag3)  +Ag2* jSU.rec[a1]/(Ag4)
  jVA0[a] <- VA0[t,s,a]
 VA0[t,s,a] <-  VA0[t,s,a] -Ag1*  VA0[t,s,a]/(Ag3) +Ag2* jVA0[a1]/(Ag4)

  jVAef1[a] <- VAef1[t,s,a]
VAef1[t,s,a] <- VAef1[t,s,a] -Ag1* VAef1[t,s,a]/(Ag3) +Ag2*jVAef1[a1]/(Ag4)
  jVAef2[a] <- VAef2[t,s,a]
VAef2[t,s,a] <- VAef2[t,s,a] -Ag1* VAef2[t,s,a]/(Ag3) +Ag2*jVAef2[a1]/(Ag4)

            for(d in 1:dd)         {
   jeEX[d,a] <- eEX[t,d,s,a]
 eEX[t,d,s,a] <-   eEX[t,d,s,a] -Ag1*  eEX[t,d,s,a]/(Ag3)  +Ag2*  jeEX[d,a1]/(Ag4)
  jeI0[d,a] <- eI0[t,d,s,a]
 eI0[t,d,s,a] <-   eI0[t,d,s,a] -Ag1*  eI0[t,d,s,a]/(Ag3)  +Ag2*  jeI0[d,a1]/(Ag4)
  jeIM0[d,a] <- eIM0[t,d,s,a]
eIM0[t,d,s,a] <-  eIM0[t,d,s,a] -Ag1* eIM0[t,d,s,a]/(Ag3)  +Ag2* jeIM0[d,a1]/(Ag4)
  jeIS0[d,a] <- eIS0[t,d,s,a]
eIS0[t,d,s,a] <-  eIS0[t,d,s,a] -Ag1* eIS0[t,d,s,a]/(Ag3)  +Ag2* jeIS0[d,a1]/(Ag4)
  jeISq[d,a] <- eISq[t,d,s,a]
eISq[t,d,s,a] <-  eISq[t,d,s,a] -Ag1* eISq[t,d,s,a]/(Ag3)  +Ag2* jeISq[d,a1]/(Ag4)
  jeISh[d,a] <- eISh[t,d,s,a]
eISh[t,d,s,a] <-  eISh[t,d,s,a] -Ag1* eISh[t,d,s,a]/(Ag3)  +Ag2* jeISh[d,a1]/(Ag4)
  jeQS[d,a] <- eQS[t,d,s,a]
 eQS[t,d,s,a] <-   eQS[t,d,s,a] -Ag1*  eQS[t,d,s,a]/(Ag3)  +Ag2*  jeQS[d,a1]/(Ag4)
  jeHS[d,a] <- eHS[t,d,s,a]
 eHS[t,d,s,a] <-   eHS[t,d,s,a] -Ag1*  eHS[t,d,s,a]/(Ag3)  +Ag2*  jeHS[d,a1]/(Ag4)

  jeEXv1[d,a] <- eEXv1[t,d,s,a]
 eEXv1[t,d,s,a] <-   eEXv1[t,d,s,a] -Ag1*  eEXv1[t,d,s,a]/(Ag3)  +Ag2*  jeEXv1[d,a1]/(Ag4)
  jeI0v1[d,a] <- eI0v1[t,d,s,a]
 eI0v1[t,d,s,a] <-   eI0v1[t,d,s,a] -Ag1*  eI0v1[t,d,s,a]/(Ag3)  +Ag2*  jeI0v1[d,a1]/(Ag4)
  jeEXv2[d,a] <- eEXv2[t,d,s,a]
 eEXv2[t,d,s,a] <-   eEXv2[t,d,s,a] -Ag1*  eEXv2[t,d,s,a]/(Ag3)  +Ag2*  jeEXv2[d,a1]/(Ag4)
  jeI0v2[d,a] <- eI0v2[t,d,s,a]
 eI0v2[t,d,s,a] <-   eI0v2[t,d,s,a] -Ag1*  eI0v2[t,d,s,a]/(Ag3)  +Ag2*  jeI0v2[d,a1]/(Ag4)

  jeIM0v[d,a] <- eIM0v[t,d,s,a]
eIM0v[t,d,s,a] <-  eIM0v[t,d,s,a] -Ag1* eIM0v[t,d,s,a]/(Ag3)  +Ag2* jeIM0v[d,a1]/(Ag4)
  jeIS0v[d,a] <- eIS0v[t,d,s,a]
eIS0v[t,d,s,a] <-  eIS0v[t,d,s,a] -Ag1* eIS0v[t,d,s,a]/(Ag3)  +Ag2* jeIS0v[d,a1]/(Ag4)
  jeISqv[d,a] <- eISqv[t,d,s,a]
eISqv[t,d,s,a] <-  eISqv[t,d,s,a] -Ag1* eISqv[t,d,s,a]/(Ag3)  +Ag2* jeISqv[d,a1]/(Ag4)
  jeIShv[d,a] <- eIShv[t,d,s,a]
eIShv[t,d,s,a] <-  eIShv[t,d,s,a] -Ag1* eIShv[t,d,s,a]/(Ag3)  +Ag2* jeIShv[d,a1]/(Ag4)
  jeQSv[d,a] <- eQSv[t,d,s,a]
 eQSv[t,d,s,a] <-   eQSv[t,d,s,a] -Ag1*  eQSv[t,d,s,a]/(Ag3)  +Ag2*  jeQSv[d,a1]/(Ag4)
```

```
                jeHSv[d,a] <- eHSv[t,d,s,a]
               eHSv[t,d,s,a] <-  eHSv[t,d,s,a] -Ag1*  eHSv[t,d,s,a]/(Ag3)  +Ag2*  jeHSv[d,a1]/(Ag4)
                jeInf[d,a] <- eInf[t,d,s,a]
               eInf[t,d,s,a] <-  eInf[t,d,s,a] -Ag1*  eInf[t,d,s,a]/(Ag3)  +Ag2*  jeInf[d,a1]/(Ag4)
                jeInfv[d,a] <- eInfv[t,d,s,a]
               eInfv[t,d,s,a] <-  eInfv[t,d,s,a] -Ag1* eInfv[t,d,s,a]/(Ag3)  +Ag2* jeInfv[d,a1]/(Ag4)
                                        }   # end d in 1:dd

                        for(v in 1:dv0)       {
                 jeVA0[v,a] <- eVA0[t,v,s,a]
               eVA0[t,v,s,a] <- eVA0[t,v,s,a] -Ag1* eVA0[t,v,s,a]/(Ag3) +Ag2*jeVA0[v,a1]/(Ag4)
                                        }  # end dt in 1:14

                        for(w in 1:tv0)       {
                 jeVAef1[w,a] <- eVAef1[t,w,s,a]
               eVAef1[t,w,s,a] <-eVAef1[t,w,s,a] -Ag1* eVAef1[t,w,s,a]/(Ag3) +Ag2* jeVAef1[w,a1]/(Ag4)
                                        }

                                tv1<-(t-vac.tim[1]); if(tv1<1) { tv1<-2  }
                        for(w in 1:tv1)       {
                 jeVAef2[w,a] <- eVAef2[t,w,s,a]
               eVAef2[t,w,s,a] <-eVAef2[t,w,s,a] -Ag1* eVAef2[t,w,s,a]/(Ag3) +Ag2* jeVAef2[w,a1]/(Ag4)
                                        }

                                 tw1 <-(t-ts); if(tw1<2) { tw1<-2  }
                        for(w in 1:tw1)       {
                   jeRE[w,a] <- eRE[t,w,s,a]
          eRE[t,w,s,a] <-   eRE[t,w,s,a] -Ag1* eRE[t,w,s,a]/(Ag3) +Ag2* jeRE[w,a1]/(Ag4)
                                        }

                        }            # end a
                        }            # end s

                        tage <- tage+365     # t for next age shifting up

                                }            # end if t==tage

#=============================================================================
#  Population estimates and all cause deaths
#   add births and minus all cause deaths according to 5yr average death rates
#=============================================================================

            for(s in 1:2)           {
            for(a in 1:10)          {

  # Monitoring deaths in SU, RE and Effectively vaccinated (non-covid deaths)
            nDth.oth[t-1,s,a] <-(SU[t-1,s,a] +RE[t-1,s,a] +SU.vac[t-1,s,a] +SU.rec[t-1,s,a] +
                                VA0[t-1,s,a] +VAef1[t-1,s,a] +VAef2[t-1,s,a]) *(rADth0[t.dr,s,a])

  # All deaths:
              nDth.all[t-1,s,a] <-nDth.oth[t-1,s,a] +nDth.cov[t-1,s,a]

#=============================================================================
              # Total numbers of S, E, I and R:
 #---------------------------------------------

               SU[t,3,11] <-   SU[t,3,11]  +  SU[t,s,a]
               EX[t,3,11] <-   EX[t,3,11]  +  EX[t,s,a]
              EXv1[t,3,11] <-  EXv1[t,3,11]  + EXv1[t,s,a]
              EXv2[t,3,11] <-  EXv2[t,3,11]  + EXv2[t,s,a]
               I0[t,3,11] <-   I0[t,3,11]  +  I0[t,s,a]
              I0v1[t,3,11] <-  I0v1[t,3,11]  + I0v1[t,s,a]
              I0v2[t,3,11] <-  I0v2[t,3,11]  + I0v2[t,s,a]
              IM0[t,3,11] <-  IM0[t,3,11]  + IM0[t,s,a]
              IS0[t,3,11] <-  IS0[t,3,11]  + IS0[t,s,a]
              ISq[t,3,11] <-  ISq[t,3,11]  + ISq[t,s,a]
              ISh[t,3,11] <-  ISh[t,3,11]  + ISh[t,s,a]
               QS[t,3,11] <-   QS[t,3,11]  +  QS[t,s,a]
               HS[t,3,11] <-   HS[t,3,11]  +  HS[t,s,a]
              IM0v[t,3,11] <-  IM0v[t,3,11]  + IM0v[t,s,a]
```

```
          IS0v[t,3,11] <-  IS0v[t,3,11]  + IS0v[t,s,a]
          ISqv[t,3,11] <-  ISqv[t,3,11]  + ISqv[t,s,a]
          IShv[t,3,11] <-  IShv[t,3,11]  + IShv[t,s,a]
           QSv[t,3,11] <-   QSv[t,3,11]  +  QSv[t,s,a]
           HSv[t,3,11] <-   HSv[t,3,11]  +  HSv[t,s,a]

            SU.vac[t,3,11] <-   SU.vac[t,3,11]  +  SU.vac[t,s,a]
            SU.rec[t,3,11] <-   SU.rec[t,3,11]  +  SU.rec[t,s,a]

           RE[t,3,11]  <- RE[t,3,11]  + RE[t,s,a]

            N[t,3,11]  <-  N[t,3,11]  +  N[t,s,a]

           HS.new[t,3,11] <-HS.new[t,3,11] +  HS.new[t,s,a]

         allHS[t] <- allHS[t] +  HS[t,s,a] + HSv[t,s,a]

          nDth.cov[t-1,3,11] <-round( nDth.cov[t-1,3,11] + nDth.cov[t-1,s,a] )

          nDthHos.all[t-1,3,11] <- nDthHos.all[t-1,3,11] +
                                   nDthHos.all[t-1,s,a] +nDthHos.allv[t-1,s,a]

          nDth.ss[t-1,3,11] <- nDth.ss[t-1,3,11] +
                                   nDth.ss[t-1,s,a] + nDth.ssv[t-1,s,a]

         nDth.oth[t-1,3,11] <- nDth.oth[t-1,3,11] +nDth.oth[t-1,s,a]

         nDth.all[t-1,3,11] <- nDth.all[t-1,3,11] +nDth.all[t-1,s,a]

         #------------------------------------------------
         # Assume the births = normal deaths to maintain the total N unchanged over time:
                  dBth[1] <- nDth.all[t-1,3,11]

         New.expal[t-1,3,11] <-New.expal[t-1,3,11] +New.expal[t-1,s,a]

         New.expsu[t-1,3,11] <-New.expsu[t-1,3,11] +New.expsu[t-1,s,a]
         New.expv0[t-1,3,11] <-New.expv0[t-1,3,11] +New.expv0[t-1,s,a]
         New.expv1[t-1,3,11] <-New.expv1[t-1,3,11] +New.expv1[t-1,s,a]
         New.expv2[t-1,3,11] <-New.expv2[t-1,3,11] +New.expv2[t-1,s,a]
         New.explosrec[t-1,3,11] <-New.explosrec[t-1,3,11] +New.explosrec[t-1,s,a]

         New.vac[t-1,3,11] <-New.vac[t-1,3,11] +New.vac[t-1,s,a]

         New.sym[t,3,11] <-round( New.sym[t,3,11] +New.sym[t,s,a] )

           VA0[t,3,11] <-   VA0[t,3,11]  +  VA0[t,s,a]
          VAef1[t,3,11] <-  VAef1[t,3,11]  + VAef1[t,s,a]
          VAef2[t,3,11] <-  VAef2[t,3,11]  + VAef2[t,s,a]

                                  }  # end a
                                  }  # end s

         # Avoiding extremely small scientic notation for these variables

            SU.vac[t,3,11] <-  round(SU.vac[t,3,11], digits=0)
            SU.rec[t,3,11] <-  round(SU.rec[t,3,11], digits=0)
              VAef1[t,3,11] <-  round(  VAef1[t,3,11], digits=0)
              VAef2[t,3,11] <-  round(  VAef2[t,3,11], digits=0)

#############################################################
##   Cumulative number of deaths
##------------------------------------------------------------
         nDthcum.cov[t] <- nDthcum.cov[t-1] +nDth.cov[t-1,3,11]

  # Sum of all infected at t:
         Infect.sum[t] <-EX[t,3,11]  +EXv1[t,3,11] +EXv2[t,3,11] +
                                  I0[t,3,11]  +I0v1[t,3,11] +I0v2[t,3,11] +
                                  IM0[t,3,11] +
                                  IS0[t,3,11] +
                                  ISq[t,3,11] +  QS[t,3,11] +
```

```
                              ISh[t,3,11] +  HS[t,3,11] +
                              IM0v[t,3,11] +
                              IS0v[t,3,11] +
                              ISqv[t,3,11] +  QSv[t,3,11] +
                              IShv[t,3,11] +  HSv[t,3,11]

   ##================================================================
   # Obtaining N[t,3,11], and so on:
   #================================================================
        for(a in 1:10)        {
        for(s in 1:2)         {
          N[t,s,11] <-  N[t,s,11] +   N[t,s,a]
          N[t,3,a] <-   N[t,3,a] +   N[t,s,a]
                              }
                              }

        t.dr <- t.dr +1                  # incremental by one day for daily death rate

        }           # end for   t in 2:tt

#------------------------------------------------------
# Output scenario specific results
#------------------------------------------------------

         Result.mul <-data.frame(Rt[], N[,3,11], SU[,3,11], SU.rec[,3,11], SU.vac[,3,11], RE[,3,11], VAef1[,3,11],
                 VAef2[,3,11], New.vac[,3,11],  New.expal[,3,11], New.sym[,3,11], HS.new[,3,11], allHS[],
                 Infect.sum[], nDth.cov[,3,11], nDthcum.cov[], nDth.all[,3,11] )
        OutFileName <- paste("...\\ResOut".out", sep="")
         write.table(Result.mul,file=OutFileName, sep="\t",quote=FALSE,append=TRUE,col.names=FALSE)
```


Song F, Bachmann MO. *BMJ Open* 2021; 11:e053507. doi: 10.1136/bmjopen-2021-053507

5.2 Input data files for running the R code

5.2-1 Age-sex-specific population, case-fatality rates, and hospitalisation rates "inParamet20.csv"

agrp	age	n_m	n_f	cft_m	cft_f	hosp_r
1	0	1679.281	1596.919	3.22E-05	1.95E-05	0.000141
2	5	1815.031	1728.698	3.22E-05	1.95E-05	0.000141
3	10	3334.701	3166.337	0.000183	0.000111	0.000516
4	20	3722.353	3549.491	0.000742	0.00045	0.014247
5	30	3776.43	3800.291	0.001806	0.001094	0.051194
6	40	3541.015	3593.888	0.003649	0.002211	0.070833
7	50	3753.483	3861.581	0.01546	0.00937	0.16
8	60	2904.167	3053.137	0.049348	0.029908	0.318919
9	70	2258.713	2521.105	0.106488	0.064538	0.535484
10	80	1163.586	1690.569	0.16573	0.100442	0.593548

5.2-2 General and household contacts per person day "inCMATRIX10.csv"

agegrp	a01	a02	a03	a04	a05	a06	a07	a08	a09	a10	h01	h02	h03	h04	h05	h06	h07	h08	h09	h10
1	1.7112	0.8034	0.6746	0.9657	1.644	0.6343	0.3257	0.1181	0.0446	0.0144	0.4788	0.5519	0.4667	0.4201	0.9003	0.188	0.06	0.0069	0.0005	0.001
2	0.6816	4.0343	0.9241	0.527	1.5545	0.9406	0.3	0.1504	0.053	0.027	0.2633	0.9183	0.6405	0.195	1.0256	0.3971	0.0324	0.0024	0.0003	0
3	0.24695	1.2525	7.7556	1.1873	1.3592	1.5294	0.5501	0.12925	0.05445	0.0531	0.13135	0.34485	1.43255	0.106	0.56165	0.624	0.0924	0.022	0.0019	0.0002
4	0.58115	0.35205	1.5978	3.92775	2.21165	1.94385	0.9814	0.1812	0.04545	0.0513	0.3288	0.1887	0.2676	0.8976	0.1759	0.25075	0.2221	0.0292	0.00135	0.0016
5	0.51325	0.91745	1.0955	1.8358	3.2517	2.0502	1.05485	0.27515	0.09215	0.0632	0.3491	0.5867	0.5529	0.0839	0.73455	0.0971	0.04185	0.019	0	0
6	0.1894	0.51185	1.8394	1.72385	2.3032	3.32855	1.1846	0.2469	0.09625	0.068	0.14665	0.3314	1.09355	0.2537	0.18465	0.5841	0.07945	0.02515	0.00905	0.0082
7	0.16105	0.16875	0.8093	1.867	1.7188	2.0792	1.86615	0.56305	0.18725	0.122	0.0701	0.05255	0.41045	0.3961	0.09315	0.10955	0.4917	0.05785	0.0032	0.0024
8	0.0431	0.23415	0.30125	0.7199	0.9967	0.9832	0.9769	0.9971	0.22895	0.1383	0.03365	0.03505	0.1736	0.1045	0.1491	0.17185	0.11215	0.5643	0.03465	0
9	0.0467	0.02455	0.4952	0.54	0.38355	0.9781	0.63465	0.8508	0.88915	1.0773	0.0103	0.0112	0.2227	0.0135	0.0207	0.25315	0.0555	0.1004	0.31395	0.3604
10	0.0206	0.0007	0.2481	0.1256	0.386	0.7299	0.4298	0.4669	1.0408	1.4766	0.0206	0	0.0362	0.0186	0.0414	0.1252	0.0766	0	0.3811	0.5412

5.2-3 Transmission related input parameters "inParamet20.csv"

paramet	t_ts	t_73	t_77	t_84	t_187	t_254	t_310	t_337	t_371	t_433	t_506	re_norm
rFSQ1	0.1	0.1	0.4	0.8	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9
rFSQ2	0.1	0.2	0.6	0.8	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95
rFSQ3	0.1	0.4	0.8	0.9	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95
a1m	4	4	4	4	4	4	4	4	4	4	4	4
a1s	2	2	2	2	2	2	2	2	2	2	2	2
a2m	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5
a2s	1.061	1.061	1.061	1.061	1.061	1.061	1.061	1.061	1.061	1.061	1.061	1.061
gm0m	5	5	5	5	5	5	5	5	5	5	5	5
gm0s	2.236	2.236	2.236	2.236	2.236	2.236	2.236	2.236	2.236	2.236	2.236	2.236
gs0m	5	5	5	5	5	5	5	5	5	5	5	5
gs0s	2.236	2.236	2.236	2.236	2.236	2.236	2.236	2.236	2.236	2.236	2.236	2.236
gs1m	4	3	3	2	2	2	2	2	2	2	2	2
gs1s	2	1.732	1.732	1.414	1.414	1.414	1.414	1.414	1.414	1.414	1.414	1.414
gs2m	2	3	3	4	4	4	4	4	4	4	4	4
gs2s	1.414	1.732	1.732	2	2	2	2	2	2	2	2	2
gh1m	4	3	3	2	2	2	2	2	2	2	2	2
gh1s	2	1.732	1.732	1.414	1.414	1.414	1.414	1.414	1.414	1.414	1.414	1.414
gh2m	10	10	10	10	10	10	10	10	10	10	10	10
gn2s	3.162	3.162	3.162	3.162	3.162	3.162	3.162	3.162	3.162	3.162	3.162	3.162
ac1a	1	0.9	0.8	0.4	0.6	0.8	0.6	0.7	0.4	0.7	0.9	1
ac3a	1	0.8	0.7	0.3	0.55	0.6	0.3	0.6	0.3	0.6	0.8	1
ac4a	1	0.7	0.6	0.2	0.4	0.5	0.2	0.5	0.2	0.5	0.7	1
ac5a	1	0.6	0.5	0.15	0.2	0.3	0.15	0.3	0.15	0.3	0.6	1

5.2-4 Population age-sex-specific death risk by week "inDTH1519Eng10.csv"

agrp	a01	a02	a03	a04	a05	a06	a07	a08	a09	a10
m01	0.002671	0.000229	0.000669	0.001443	0.002844	0.007457	0.01506	0.038052	0.097395	0.372278
m02	0.002593	0.000153	0.000627	0.00166	0.002807	0.00697	0.014316	0.038425	0.098107	0.383425
m03	0.002907	0.000229	0.00046	0.001479	0.002696	0.006708	0.01387	0.036699	0.094224	0.37748
m04	0.002593	0.000229	0.000544	0.001479	0.002881	0.006445	0.014019	0.034694	0.093447	0.368439
m05	0.002907	0.000229	0.000544	0.001407	0.00277	0.006745	0.014316	0.036233	0.091376	0.359646
m06	0.002593	0.000229	0.00046	0.001299	0.002918	0.00652	0.013647	0.035487	0.091247	0.347509
m07	0.002593	0.000229	0.000585	0.001515	0.002474	0.006033	0.013684	0.0353	0.090276	0.35333
m08	0.002593	0.000153	0.000544	0.001696	0.002511	0.005808	0.013833	0.034694	0.088529	0.338221
m09	0.002436	0.000229	0.000585	0.001515	0.002696	0.006558	0.013721	0.034648	0.088141	0.3324
m10	0.00275	0.000153	0.000544	0.001227	0.00277	0.005996	0.013498	0.034974	0.086393	0.336487
m11	0.002671	0.000153	0.000585	0.001443	0.002548	0.006258	0.013498	0.033622	0.085099	0.315062
m12	0.002907	0.000153	0.00046	0.001407	0.002474	0.00652	0.013238	0.032689	0.082899	0.313328
m13	0.002279	0.000153	0.000585	0.001588	0.002696	0.006183	0.012792	0.032736	0.083352	0.30404
m14	0.002514	0.000153	0.000544	0.001299	0.002733	0.005696	0.012494	0.033015	0.082964	0.307879
m15	0.002514	0.000229	0.000585	0.001588	0.002696	0.006183	0.013163	0.032736	0.081087	0.297724
m16	0.002671	0.000229	0.000502	0.001299	0.002622	0.005808	0.01294	0.032176	0.080116	0.294132
m17	0.002436	0.000153	0.000544	0.001588	0.002548	0.005546	0.012754	0.032036	0.078887	0.285463
m18	0.002593	0.000153	0.000585	0.001407	0.002881	0.005771	0.012717	0.031943	0.07811	0.285339
m19	0.002436	0.000153	0.000502	0.001515	0.002659	0.005883	0.012308	0.031523	0.078692	0.283977
m20	0.002514	0.000153	0.000418	0.001335	0.002585	0.005808	0.012606	0.031104	0.076686	0.276794
m21	0.002671	7.64E-05	0.000502	0.001479	0.002659	0.005733	0.012643	0.03129	0.075198	0.276051
m22	0.0022	0.000306	0.000627	0.001696	0.002474	0.005321	0.012271	0.030451	0.074162	0.265029
m23	0.002514	0.000306	0.000418	0.001515	0.002511	0.005471	0.012159	0.030264	0.076686	0.264286
m24	0.002593	0.000306	0.000544	0.001299	0.002659	0.005996	0.012234	0.029705	0.074551	0.262552
m25	0.002279	0.000153	0.000544	0.001371	0.002437	0.005621	0.012457	0.030217	0.076104	0.266886
m26	0.002514	0.000229	0.000502	0.001263	0.002585	0.005471	0.011862	0.030264	0.07468	0.25896
m27	0.002279	0.000153	0.000418	0.001551	0.002696	0.005359	0.012568	0.030684	0.075716	0.266639
m28	0.002357	0.000153	0.000544	0.001479	0.002327	0.005584	0.011899	0.029891	0.072415	0.255988
m29	0.002671	0.000229	0.000418	0.001371	0.002511	0.005659	0.012978	0.029984	0.073127	0.262799
m30	0.002436	0.000229	0.000418	0.001263	0.002548	0.005771	0.01149	0.029098	0.07578	0.262056
m31	0.002279	0.000153	0.000544	0.001551	0.002511	0.005584	0.011936	0.028958	0.071703	0.25252
m32	0.002436	0.000306	0.000544	0.001371	0.002659	0.005846	0.012085	0.030404	0.073062	0.262304
m33	0.00275	0.000229	0.000502	0.001299	0.002548	0.005621	0.011862	0.029425	0.074227	0.261313
m34	0.002121	7.64E-05	0.000376	0.001299	0.002511	0.004797	0.012011	0.029378	0.073127	0.260322
m35	0.002907	0.000153	0.000502	0.001588	0.002401	0.005021	0.012383	0.030404	0.071833	0.257226
m36	0.001964	0.000153	0.000585	0.001371	0.002179	0.005434	0.011564	0.028539	0.07481	0.256359
m37	0.002279	0.000153	0.000376	0.001335	0.002401	0.005171	0.011713	0.030031	0.07591	0.268992
m38	0.002043	0.000229	0.000335	0.001335	0.002548	0.005359	0.012494	0.029425	0.073515	0.270354
m39	0.002671	0.000153	0.000418	0.001191	0.002511	0.005246	0.011974	0.029891	0.076427	0.276546
m40	0.002436	0.000229	0.000585	0.001515	0.002437	0.005471	0.012234	0.030031	0.079534	0.278528
m41	0.002436	0.000153	0.000585	0.001371	0.002437	0.005509	0.012606	0.030357	0.079598	0.288435
m42	0.002514	0.000153	0.000585	0.001407	0.00229	0.005621	0.011788	0.03087	0.078239	0.294008
m43	0.002671	0.000229	0.000502	0.001299	0.00229	0.005359	0.012457	0.030964	0.078175	0.290417
m44	0.002514	0.000229	0.000585	0.001227	0.002364	0.005509	0.012271	0.031803	0.081475	0.296609
m45	0.002357	0.000229	0.000502	0.001118	0.002364	0.005284	0.012234	0.032502	0.081669	0.303421
m46	0.002514	0.000229	0.00046	0.001443	0.002401	0.005584	0.012643	0.031383	0.082058	0.305402
m47	0.00275	0.000153	0.000627	0.001227	0.002364	0.005621	0.012606	0.03171	0.082511	0.306145
m48	0.002671	7.64E-05	0.00046	0.001443	0.002327	0.005434	0.013387	0.031756	0.080699	0.313824
m49	0.002357	0.000306	0.000502	0.001335	0.002548	0.005434	0.013238	0.032782	0.085293	0.322369
m50	0.002121	0.000306	0.000585	0.001371	0.002511	0.005471	0.013163	0.033388	0.086005	0.327199
m51	0.002671	0.000306	0.000418	0.001299	0.002511	0.005584	0.013387	0.034135	0.088723	0.336859
m52	0.002593	0.000382	0.00046	0.001299	0.002474	0.006108	0.013535	0.034508	0.089694	0.347509

f01	0.002314	0.00024	0.000308	0.000713	0.001725	0.004231	0.010467	0.026299	0.068384	0.358013
f02	0.002562	0.000401	0.000308	0.000638	0.001652	0.003936	0.009381	0.025145	0.069481	0.368451
f03	0.001901	0.00024	0.000352	0.000713	0.001725	0.0039	0.009344	0.024303	0.06821	0.364013
f04	0.002396	0.00016	0.00022	0.000676	0.001432	0.003863	0.00891	0.023771	0.063301	0.347739
f05	0.002149	0.00024	0.000352	0.000638	0.001578	0.003863	0.009417	0.023904	0.063994	0.342151
f06	0.002644	0.00016	0.000352	0.000751	0.001578	0.003863	0.00891	0.023061	0.062897	0.322343
f07	0.002066	0.00016	0.000308	0.000826	0.001468	0.004047	0.009018	0.02346	0.062261	0.320864
f08	0.001983	0.00016	0.000352	0.000788	0.001578	0.00401	0.009018	0.022706	0.061626	0.315768
f09	0.002231	0.00016	0.000308	0.000788	0.001615	0.00401	0.008801	0.023194	0.061453	0.311412
f10	0.002066	0.00024	0.000396	0.000638	0.001432	0.00401	0.009091	0.022041	0.060818	0.308618
f11	0.001901	8.02E-05	0.000308	0.000713	0.001542	0.003789	0.008729	0.022041	0.06024	0.301221
f12	0.002149	0.00016	0.000352	0.000826	0.001395	0.003716	0.008366	0.02142	0.058681	0.286345
f13	0.001983	0.00024	0.000264	0.000751	0.001615	0.003458	0.008692	0.022085	0.057872	0.280427
f14	0.001901	0.00016	0.000308	0.000638	0.001468	0.003789	0.00862	0.021287	0.056024	0.275907
f15	0.001983	0.00024	0.000352	0.000713	0.001468	0.003826	0.008475	0.021198	0.056428	0.273194
f16	0.002066	0.00016	0.000352	0.000676	0.001505	0.003605	0.008765	0.02111	0.055331	0.267195
f17	0.002066	0.00016	0.000308	0.000676	0.001468	0.003789	0.00804	0.02142	0.053771	0.257414
f18	0.001735	0.00024	0.000264	0.000638	0.001395	0.003532	0.008149	0.020888	0.053078	0.253798
f19	0.002066	0.00016	0.000308	0.000601	0.001395	0.003752	0.008547	0.020445	0.052963	0.254291
f20	0.001983	0.00016	0.000352	0.000751	0.001578	0.003679	0.007968	0.020445	0.051865	0.248209
f21	0.001901	0.00016	0.000264	0.000751	0.001505	0.003642	0.00804	0.020134	0.051923	0.246976
f22	0.002149	0.00016	0.000264	0.000676	0.001468	0.003421	0.007932	0.019602	0.051403	0.237689
f23	0.001901	8.02E-05	0.000308	0.000488	0.001468	0.003421	0.007968	0.020622	0.051634	0.234319
f24	0.002314	0.00016	0.000396	0.000563	0.001358	0.003495	0.008258	0.019292	0.050883	0.237032
f25	0.001983	0.00016	0.000176	0.000676	0.001505	0.003679	0.007714	0.019602	0.050826	0.238429
f26	0.001653	8.02E-05	0.000396	0.000676	0.001468	0.003752	0.008185	0.019691	0.049266	0.231114
f27	0.002066	0.00024	0.00044	0.000638	0.001395	0.003458	0.008511	0.019025	0.051865	0.237114
f28	0.002149	0.00016	0.000352	0.000563	0.001432	0.003605	0.007896	0.018493	0.049959	0.225854
f29	0.001735	8.02E-05	0.000264	0.000638	0.001432	0.003458	0.007859	0.019114	0.050017	0.235059
f30	0.002066	0.00024	0.000308	0.000525	0.001725	0.003495	0.007751	0.019513	0.050537	0.235059
f31	0.002231	0.00016	0.000264	0.000638	0.001505	0.003532	0.008004	0.018538	0.048689	0.223553
f32	0.002149	8.02E-05	0.000352	0.000525	0.001432	0.003348	0.008113	0.019513	0.049035	0.23136
f33	0.002066	8.02E-05	0.000308	0.000563	0.001395	0.003458	0.007896	0.018892	0.050306	0.230867
f34	0.001735	0.00016	0.000352	0.000713	0.001358	0.003201	0.007533	0.020045	0.049959	0.231278
f35	0.002231	8.02E-05	0.00022	0.000751	0.001395	0.003532	0.007896	0.018937	0.050364	0.226758
f36	0.001818	0.00024	0.000264	0.000563	0.001285	0.003421	0.007751	0.018759	0.050537	0.22947
f37	0.002066	0.00016	0.000308	0.000638	0.001358	0.003348	0.008222	0.019779	0.050999	0.229635
f38	0.001901	0.00024	0.00022	0.000601	0.001285	0.003348	0.008294	0.019646	0.051519	0.236621
f39	0.002314	0.00016	0.000264	0.000601	0.001468	0.003458	0.008403	0.019558	0.053194	0.245004
f40	0.002231	0.00024	0.000352	0.000525	0.001321	0.003642	0.007896	0.020001	0.052327	0.245743
f41	0.002396	8.02E-05	0.000352	0.000601	0.001248	0.003495	0.008222	0.019957	0.053713	0.259222
f42	0.001901	0.00024	0.000264	0.000638	0.001395	0.003421	0.008004	0.02009	0.054176	0.259962
f43	0.002231	0.00024	0.00022	0.000563	0.001358	0.003458	0.00833	0.020533	0.05406	0.25725
f44	0.002231	0.00016	0.00022	0.000488	0.001615	0.003458	0.008801	0.020533	0.054233	0.267852
f45	0.002231	0.00024	0.000396	0.000488	0.001432	0.003532	0.008801	0.020489	0.055619	0.270236
f46	0.002149	0.000321	0.000396	0.000525	0.001395	0.003495	0.008801	0.020711	0.0551	0.270153
f47	0.001818	0.00016	0.000308	0.000563	0.001468	0.003789	0.008366	0.021553	0.057525	0.269332
f48	0.002066	0.00024	0.00022	0.000638	0.001505	0.0039	0.008403	0.021198	0.057352	0.279359
f49	0.001983	0.00016	0.000264	0.000563	0.001432	0.003789	0.00891	0.021598	0.061511	0.290536
f50	0.001901	0.00024	0.000352	0.000563	0.001468	0.004047	0.008366	0.022751	0.061164	0.298262
f51	0.002314	0.00024	0.000352	0.000563	0.001505	0.003495	0.009091	0.023416	0.062435	0.311823
f52	0.001901	0.00016	0.000352	0.000638	0.001615	0.003789	0.009055	0.023061	0.064687	0.316097

6. References for supplementary material

1. Lavine JS, Bjornstad ON, Antia R. Immunological characteristics govern the transition of COVID-19 to endemicity. *Science (New York, NY)* 2021;371(6530):741-45. doi: 10.1126/science.abe6522 [published Online First: 2021/01/14]

2. Olliaro P. What does 95% COVID-19 vaccine efficacy really mean? *Lancet Infect Dis* 2021 doi: 10.1016/s1473-3099(21)00075-x [published Online First: 2021/02/21]

3. Rock K, Brand S, Moir J, et al. Dynamics of infectious diseases. *Rep Prog Phys* 2014;77(2):026602. doi: 10.1088/0034-4885/77/2/026602 [published Online First: 2014/01/22]

4. Tuckwell HC, Williams RJ. Some properties of a simple stochastic epidemic model of SIR type. *Math Biosci* 2007;208(1):76-97. doi: 10.1016/j.mbs.2006.09.018 [published Online First: 2006/12/19]

5. Grant A. Dynamics of COVID-19 epidemics: SEIR models underestimate peak infection rates and overestimate epidemic duration. *medRxiv preprint doi: https://doiorg/101101/2020040220050674* 2020

6. Krylova O, Earn DJD. Effects of the infectious period distribution on predicted transitions in childhood disease dynamics. *J R Soc Interface* 2013;10(84):20130098-98. doi: 10.1098/rsif.2013.0098

7. Davies NG, Kucharski AJ, Eggo RM, et al. Effects of non-pharmaceutical interventions on COVID-19 cases, deaths, and demand for hospital services in the UK: a modelling study. *The Lancet Public health* 2020;5(7):e375-e85. doi: 10.1016/s2468-2667(20)30133-x [published Online First: 2020/06/06]

8. GOV.UK. Coronavirus (COVID-19) in the UK 2021 [Available from: https://coronavirus.data.gov.uk/ accessed 31 July 2021.

9. Davies NG, Klepac P, Liu Y, et al. Age-dependent effects in the transmission and control of COVID-19 epidemics. *Nature medicine* 2020;26(8):1205-11. doi: 10.1038/s41591-020-0962-9 [published Online First: 2020/06/18]

10. Zhang J, Litvinova M, Wang W, et al. Evolving epidemiology and transmission dynamics of coronavirus disease 2019 outside Hubei province, China: a descriptive and modelling study. *Lancet Infect Dis* 2020 doi: 10.1016/s1473-3099(20)30230-9 [published Online First: 2020/04/06]

11. Ferguson N, Laydon D, Nedjati-Gilani G, et al. Impact of non-pharmaceutical interventions (NPIs) to reduce COVID-19 mortality and healthcare demand.: Imperial College COVID-19 Response Team., 2020.

12. Verity R, Okell LC, Dorigatti I, et al. Estimates of the severity of coronavirus disease 2019: a model-based analysis. *Lancet Infect Dis* 2020 doi: 10.1016/s1473-3099(20)30243-7 [published Online First: 2020/04/03]

13. Chinese Center of Disease Prevention and Control. [The epidemiological characteristics of an outbreak of 2019 novel coronavirus diseases (COVID-19) in China]. *Zhonghua Liu Xing Bing Xue Za Zhi* 2020;41(2):145-51. doi: 10.3760/cma.j.issn.0254-6450.2020.02.003 [published Online First: 2020/02/18]

14. Government Office for Science and Scientific Advisory Group for Emergencies. Guidance: The R number and growth rate in the UK. From Government Office for Science and Scientific Advisory Group for Emergencies. 2020 [updated Last updated 2 October 2020. Available from: https://www.gov.uk/guidance/the-r-number-in-the-uk#contents accessed 08/10/2020.

15. Gov.uk. Coronavirus (COVID-19): Guidance and support. 2021 [Available from: https://www.gov.uk/coronavirus accessed 30 July 2021.

16. Rubin D, Huang J, Fisher BT, et al. Association of Social Distancing, Population Density, and Temperature With the Instantaneous Reproduction Number of SARS-CoV-2 in Counties Across the United States. *JAMA network open* 2020;3(7):e2016099. doi: 10.1001/jamanetworkopen.2020.16099 [published Online First: 2020/07/24]

17. Li F, Li YY, Liu MJ, et al. Household transmission of SARS-CoV-2 and risk factors for susceptibility and infectivity in Wuhan: a retrospective observational study. *Lancet Infect Dis* 2021;21(5):617-28. doi: 10.1016/s1473-3099(20)30981-6 [published Online First: 2021/01/22]

18. Mossong J, Hens N, Jit M, et al. Social contacts and mixing patterns relevant to the spread of infectious diseases. *PLoS Med* 2008;5(3):e74. doi: 10.1371/journal.pmed.0050074 [published Online First: 2008/03/28]

19. PHE. COVID-19 vaccine surveillance report Week 30 Public Health England 2021.

20. Le Bert N, Tan AT, Kunasegaran K, et al. SARS-CoV-2-specific T cell immunity in cases of COVID-19 and SARS, and uninfected controls. *Nature* 2020;584(7821):457-62. doi: 10.1038/s41586-020-2550-z [published Online First: 2020/07/16]

21. Widge AT, Rouphael NG, Jackson LA, et al. Durability of Responses after SARS-CoV-2 mRNA-1273 Vaccination. *The New England journal of medicine* 2021;384(1):80-82. doi: 10.1056/NEJMc2032195 [published Online First: 2020/12/04]

22. Mallapaty S. COVID vaccines slash viral spread - but Delta is an unknown. *Nature* 2021 doi: 10.1038/d41586-021-02054-z [published Online First: 2021/07/30]

23. Gov.uk. Slides and datasets to accompany coronavirus press conference: 5 June 2020. Published 5 June 2020 ed, 2020.

24. Prem K, Cook AR, Jit M. Projecting social contact matrices in 152 countries using contact surveys and demographic data. *PLoS Comput Biol* 2017;13(9):e1005697. doi: 10.1371/journal.pcbi.1005697 [published Online First: 2017/09/13]

25. Office for National Statistics. Dataset: 2018-based National population projections table of contents, 2018.

26. Mahase E. Covid-19: What have we learnt about the new variant in the UK? *Bmj* 2020;371:m4944. doi: 10.1136/bmj.m4944 [published Online First: 2020/12/29]

27. Giordano G, Blanchini F, Bruno R, et al. Modelling the COVID-19 epidemic and implementation of population-wide interventions in Italy. *Nature medicine* 2020;26(6):855-60. doi: 10.1038/s41591-020-0883-7 [published Online First: 2020/04/24]

28. Gov.uk. Official Statistics. UK Biobank COVID-19 antibody study: round 1 results. : Department of Health & Social Care [Available from: https://www.gov.uk/government/publications/uk-biobank-covid-19-seroprevalence-study-round-1-results/uk-biobank-covid-19-seroprevalence-study-round-1-results accessed Published 30 July 2020. .

29. Bingham K. The UK Government's Vaccine Taskforce: strategy for protecting the UK and the world. *Lancet* 2020 doi: 10.1016/s0140-6736(20)32175-9 [published Online First: 2020/10/31]

30. Department of Health & Social Care JCVI statement, August 2021: COVID-19 vaccination of children and young people aged 12 to 17 years.: Department of Health & Social Care 2021 [Available from: https://www.gov.uk/government/organisations/department-of-health-and-social-care.

31. Mahase E. Covid-19: UK approves Pfizer and BioNTech vaccine with rollout due to start next week. 2020;371:m4714. doi: 10.1136/bmj.m4714 %J BMJ

32. PHE. COVID-19 vaccine surveillance report Week 28: Public Health England 2021.

33. Hall ea. Effectiveness of BNT162b2 mRNA Vaccine Against Infection and COVID-19 Vaccine Coverage in Healthcare Workers in England, Multicentre Prospective Cohort Study (the SIREN Study). *Lancet (preprint)* 2021

34. Ramsay M. COVID-19: analysing first vaccine effectiveness in the UK 2021 [updated 23 February 2021. Available from: https://publichealthmatters.blog.gov.uk/2021/02/23/covid-19-analysing-first-vaccine-effectiveness-in-the-uk/.

35. Department of Health and Social Care. Optimising the covid-19 vaccination programme for maximum short-term impact. 2021 [Available from: https://www.gov.uk/government/publications/prioritising-the-first-covid-19-vaccine-dose-jcvi-statement/optimising-the-covid-19-vaccination-programme-for-maximum-short-term-impact.

36. Levine-Tiefenbrun M, Yelin I, Katz R, et al. Decreased SARS-CoV-2 viral load following vaccination. 2021:2021.02.06.21251283. doi: 10.1101/2021.02.06.21251283 %J medRxiv

37. Emary KRW, Golubchik T, Aley PK, et al. Efficacy of ChAdOx1 nCoV-19 (AZD1222) Vaccine Against SARS-CoV-2 VOC 202012/01 (B.1.1.7). SSRN [Preprint]. 2021. DOI: 10.2139/ssrn.3779160. *Available at: https://papersssrncom/sol3/paperscfm?abstract_id=3779160* 2021

38. Liu QH, Ajelli M, Aleta A, et al. Measurability of the epidemic reproduction number in data-driven contact networks. *Proceedings of the National Academy of Sciences of the United States of America* 2018;115(50):12680-85. doi: 10.1073/pnas.1811115115 [published Online First: 2018/11/23]