*Appendix for*

Quantifying the seasonal driver of transmission for   
Lassa fever in Nigeria

Andrei R. Akhmetzhanov, Yusuke Asai, Hiroshi Nishiura

**Contests**

1. **Analysis of a nosocomial outbreak in Jos, Nigeria, in 1970**

*Reconstructing the timeline of the outbreak*

*Inference of model parameters using the Bayesian approach*

*MCMC iterations*

1. **Model-based inference using the human case (incidence) data**

*Maximum likelihood estimation*

*Fitting procedure using the Bayesian approach*

*MCMC iterations*

1. **Dynamic modelling of LF transmission in rodent populations**

*Model formulation*

*Periodic conditions for the change in population size*

*Susceptible-infected-recovered (SIR) model*

*Inference of model parameters*

1. **Computer simulations and code sharing**

Appendix references

Appendix figures

Appendix code snippets

**A. Analysis of a nosocomial outbreak in Jos, Nigeria, in 1970**

*Reconstructing the timeline of the outbreak*

We used the available event time data of a nosocomial outbreak in Jos, Nigeria, in 1970, to estimate the incubation period and the time from illness onset to death. In total, the outbreak involved 23 cases excluding the index case. Each case record contained a lower and an upper time boundary of the probable date of exposure to Lassa fever (LF) virus as well as two other event times, i.e., the date of illness onset and the date of death (Appendix Fig 1).

*Inference of model parameters using the Bayesian approach*

In our Markov Chain Monte Carlo (MCMC) iterations, we employed a normal distribution for the prior distribution of each exposure time, :

The mean and standard deviation (sd) were used to ensure that the majority of values belonged to the interval between and The normal distribution was preferred to the uniform prior to avoid the edge effects from the boundaries [1], see also discussion in [2].

Then we defined two main distributions for the incubation period and the time from illness onset to death, and employed the gamma distribution for each:

where hyper-parameters, the means and , and standard deviations and , have non-informative positively defined priors given by a half flat (positive) distribution with arbitrarily small shape and scale parameters:

The observed times of illness onset and death were subsequently inferred using sampling from normal distributions with a fixed standard deviation 0.5, which correlates with a half of the time scale in the timeline as a source of measurement error:

*MCMC iterations*

We performed MCMC iterations with 500,000 iterations, plus 20,000 iterations as a burn-in period. The thinning parameter of and was equal to 50, and the exposure time was set at 10. The resulting trace plots for the mean and variance complemented with autocorrelation plots shown in Appendix Figures 2–5 demonstrate the sufficient convergence power of the iterative process.

**B. Model-based inference using the human case (incidence) data**

*Maximum likelihood estimation*

As stated in the main text, we used maximum likelihood estimation (MLE) to obtain point estimates of model parameters Specifically, we maximized the total (composite) likelihood of the form:

with respect to three varied parameters for a fixed pair of indices . Then we performed a grid search over all possible values spanning the range to determine the global maximum of the likelihood Each point estimate can be complimented with 95% confidence intervals, e.g., based on the likelihood profile.

Appendix Figure 6 shows the resulting fit using the MLE procedure. For comparison, the fit of the analogous model with a (single) constant exposure rate is shown in Appendix Figure 7. The single exposure rate model yielded a greater Akaike information criterion value (3679.2 in contrast to 2877.2 with two exposure model).

*Fitting procedure using the Bayesian approach*

We found the level of uncertainty in each model parameter estimate by conducting MCMC simulations for Bayesian inference. Here, we describe the sampling procedure for each estimate by inferring an underlying posterior distribution. Specifically, we were interested in estimating the posterior of the exposure rate *a* for each high/low-risk period, and the CFR *q*.

The exposure rate *aw* at week number *w* was sampled from the Gamma distribution:

where is a corresponding calendar week, whereas a pair of hyper-parameters and are given by the following expressions:

We supplemented each with the following non-informative prior distributions: that are half flat (positive) distributions. The time boundaries had the uniform prior distributions: which ensure: We also imposed constraints: and

The risk of death *q* is sampled from a Gamma distribution to ensure the positive range of its values:

where two hyper-parameters have the non-informative prior distributions of the form: that are analogous to the priors for the exposure hyper-parameters used above.

To infer parameters, we sampled the number of cases and the number of deaths from Poisson and Binomial distributions, respectively:

where the rates and resulted from convolution sums. The first being defined by the formula:

where denotes the incubation time distribution, see Appendix A. Whereas, the second rate was given by the following formula:

where is the distribution of time periods from exposure to death. In turn, we convoluted the distribution with the distribution of time periods from illness onset to death obtained earlier in Appendix A, i.e.:

*MCMC iterations*

To obtain posterior distributions of model parameters we compiled MCMC iterations with 18 chains, each characterized by 100,000 iterations and a burn-in period consisting of 20,000 iterations (Appendix Code Snippets). The thinning parameter for all parameters was set to 100 to avoid correlation effects in the chain. The resulting trace and density plots, as well as the autocorrelation plots for each pair of parameters, are shown in Appendix Figures 8–16. As shown, a sufficient level of convergence of the iterative algorithm was obtained.

**C. Dynamic modelling of LF transmission in rodent populations**

*Model formulation*

To model the transmission dynamics of LF in rodents, we adopted the modelling framework reported by Peel *et al.* [3]. We let be the population size of rodents at time *t* ( – where the season length is scaled to one). Then, the dynamics could be represented as follows:

where is the mortality rate *per capita*, and the growth rate *per capita*  is given by a periodic Gaussian function: The mortality rate for rodent species *Mastomys natalensis* is remarkably high: whereas the two parameters and were previously identified from 20-year observations of rodent species in Tanzania: see [4] and Supplementary Figure 1 therein. To adjust the model for a situation in Nigeria, we considered a lower average population density of 60 rodents per Ha compared to 80 rodents per Ha as in [4]. Whereas the seasonality parameter will be the subject of the model fit.

Due to apparent differences in climate between Tanzania and Nigeria, such as changes in the timing and length of the rainy season, a time shift was imposed to the function to obtain the seasonal population dynamics in Nigeria. Specifically, we first identified the end of the rainy season in both countries based on historical averages of rainfall over the time period 1901–2015: for part of the year for Tanzania, for part of the year for Nigeria (Appendix Figure 16). Then we calculated the difference between these two times as for part of the year, and accounted for the shift in the birth rate function of this value: (Fig S3*b*).

*Periodic conditions for the change in population size*

We chose a parameter *κ* to satisfy the periodicity condition for We required: which translates to:

This rewrites as follows:

to obtain:  Here, denotes the Bessel function of the first kind. Thus, we did not need to assign an estimate for parameter .

We also note that the average population density can be defined as follows:

*Susceptible-infected-recovered (SIR) model*

The flow diagram (Fig S3*a*) translates into the following dynamical equations:

for the three components: susceptible infected and recovered respectively. All rates are defined as *per capita*, and denote transition rates between compartments and and and respectively. Two other rates and are the rates of vertical (parental) transmission of the LF virus and its antibodies, respectively. The transmission coefficient is set to a variable , which consists of the rate at which a susceptible rodent becomes infectious when in contact with an infected rodent, and contact-density function The latter has been previously found by fitting the observed experimental data of contact pattern in rodents to the sigmoidal function [4,5]:

We translated the absolute quantities of infected and recovered rodents to their densities with respect to the total size of the population. We also introduced two new variables: and , and redefined the system of dynamic equations written above as follows:

|  |  |  |
| --- | --- | --- |
|  |  | (S1) |

The number of susceptible rodents is given by:

*Inference of model parameters*

To predict the transmission dynamics of LF infection among rodents, we use documented facts on the course of infection [6]: (i) LF infection is asymptomatic in rodents; (ii) it is also transient, with the virus being cleared from the blood after the infection; (iii) LF virus can be secreted in urine up to 103 days; (iv) there is a probability of horizontal and vertical transmission of LF virus and its antibodies. This results in the following choice of parameters, see Fig S3*c*. In particular, we define the characteristic recovery time from the infection as 90 days, and the rate of recovery is therefore set to: The characteristic time at which immunity is lost is assigned as 120 days, i.e., the respective rate is set to:

However, we still do not know the rate between susceptible and infected rodents that characterizes the transmission dynamics in the SIR model, the seasonality parameter and possibly unknown average population size Identification of this becomes a subject of the model fit.

To determine the values of model parameters, we required the mean prevalence levels of LF infection among rodents in the dry and rainy seasons, defined as and to be close to the observed levels in the field experiments. Fichet–Calvet and colleagues [6] reported that a one-time measurement in the dry season revealed the mean prevalence to be at 29%, while five independent measurements (one in the dry season and four in the rainy season) showed the mean level of prevalence to be 43%. Hence, we adjusted the mean prevalence level separately for both the dry and rainy seasons by using a simple algebraic rule: if is the mean prevalence level observed in the rainy season, then it should satisfy the following condition for five independent measurements: This yields the value: Thus, we required the values and predicted by our model to be close to 29% and 46.5%, respectively.

A fitting procedure was carried out by using the *χ*2 statistic. Specifically, we tried to minimize the quantity:

where and are values obtained from the dynamics (S1). Appendix Figure 18 shows that the minimum of is reached on some one-dimensional manifold in the space of two parameters and The fitted dynamics and the evaluated values of and for fixed are shown in Figure S3.

**D. Computer simulations and code sharing**

All calculations were made using free, open-source statistical and programming environments (*R* Version 3.5.1, *Python* Version 3.6.6, and *Julia* Version 1.0.1). The results were tested in multiple computational environments to ensure the validity of the calculations and to avoid processing errors. MCMC simulations were performed using the *R* package NIMBLE Version 0.6-12 [7], and comparative analysis of different packages can be found elsewhere [8]. To run the significance test of association between the observed LF incidence and climatological variables, we used the *R* package rEDM [9]. The code for all calculations and to reproduce all of the figures are accessible from the open-shared repository: http://tiny.cc/Lassa18Scripts. This information can be used freely for non-commercial purposes.

Appendix References

1. Carpenter B. 2017 Computational and statistical issues with uniform interval priors. Available at: andrewgelman.com/2017/11/28/computational-statistical-issues-uniform-interval-priors/. Accessed on August 30, 2018.

2. Park SW, Dushoff J, Earn DJD, Poinar H, Bolker BM. 2018 Human ectoparasite transmission of the plague during the Second Pandemic is only weakly supported by proposed mathematical models. *Proc. Natl. Acad. Sci.* **115**, E7892–E7893. (doi:10.1073/pnas.1809775115)

3. Peel AJ, Pulliam JRC, Luis AD, Plowright RK, O’Shea TJ, Hayman DTS, Wood JLN, Webb CT, Restif O. 2014 The effect of seasonal birth pulses on pathogen persistence in wild mammal populations. *Proc. R. Soc. B Biol. Sci.* **281**, 20132962–20132962. (doi:10.1098/rspb.2013.2962)

4. Borremans B, Reijniers J, Hens N, Leirs H. 2017 The shape of the contact–density function matters when modelling parasite transmission in fluctuating populations. *R. Soc. Open Sci.* **4**, 171308. (doi:10.1098/rsos.171308)

5. Borremans B, Reijniers J, Hughes NK, Godfrey SS, Gryseels S, Makundi RH, Leirs H. 2017 Nonlinear scaling of foraging contacts with rodent population density. *Oikos* **126**, 792–800. (doi:10.1111/oik.03623)

6. Fichet-Calvet E, Becker-Ziaja B, Koivogui L, Günther S. 2014 Lassa Serology in Natural Populations of Rodents and Horizontal Transmission. *Vector-Borne Zoonotic Dis.* **14**, 665–674. (doi:10.1089/vbz.2013.1484)

7. de Valpine P, Turek D, Paciorek CJ, Anderson-Bergman C, Lang DT, Bodik R. 2017 Programming With Models: Writing Statistical Algorithms for General Model Structures With NIMBLE. *J. Comput. Graph. Stat.* **26**, 403–413. (doi:10.1080/10618600.2016.1172487)

8. Li M, Dushoff J, Bolker BM. 2018 Fitting mechanistic epidemic models to data: A comparison of simple Markov chain Monte Carlo approaches. *Stat. Methods Med. Res.* **27**, 1956–1967. (doi:10.1177/0962280217747054)

9. Ye H, Clark A, Deyle E, Munch S, et al. In press. *rEDM: Applications of Empirical Dynamic Modeling from Time Series. R package version 0.7.1.*

10. Carey DE, Kemp GE, White HA, Pinneo L, Addy RF, Fom ALMD, Stroh G, Casals J, Henderson BE. 1972 Lassa fever Epidemiological aspects of the 1970 epidemic, Jos, Nigeria. *Trans. R. Soc. Trop. Med. Hyg.* **66**, 402–408. (doi:10.1016/0035-9203(72)90271-4)

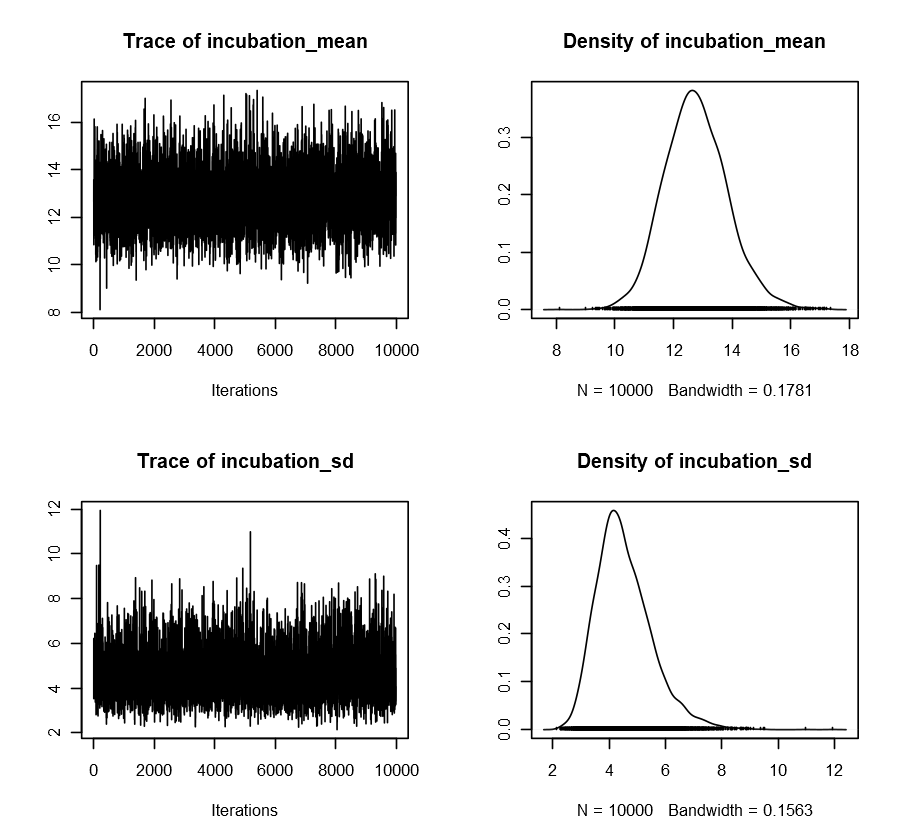
11. Sluydts V, Davis S, Mercelis S, Leirs H. 2009 Comparison of multimammate mouse (Mastomys natalensis) demography in monoculture and mosaic agricultural habitat: Implications for pest management. *Crop Prot.* **28**, 647–654. (doi:10.1016/j.cropro.2009.03.018)

12. University of East Anglia Climatic Research Unit, Jones PD, Harris IC. 2008 Climatic Research Unit (CRU) time-series datasets of variations in climate with variations in other phenomena. NCAS British Atmospheric Data Centre. Assessed August 2017.

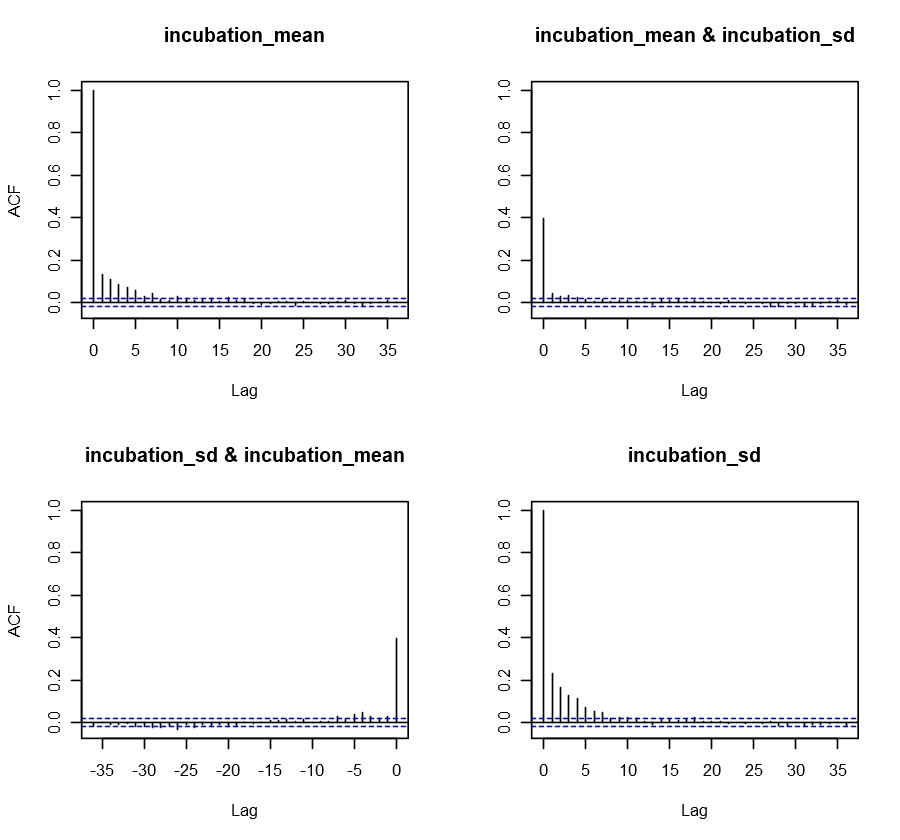
Appendix Figures



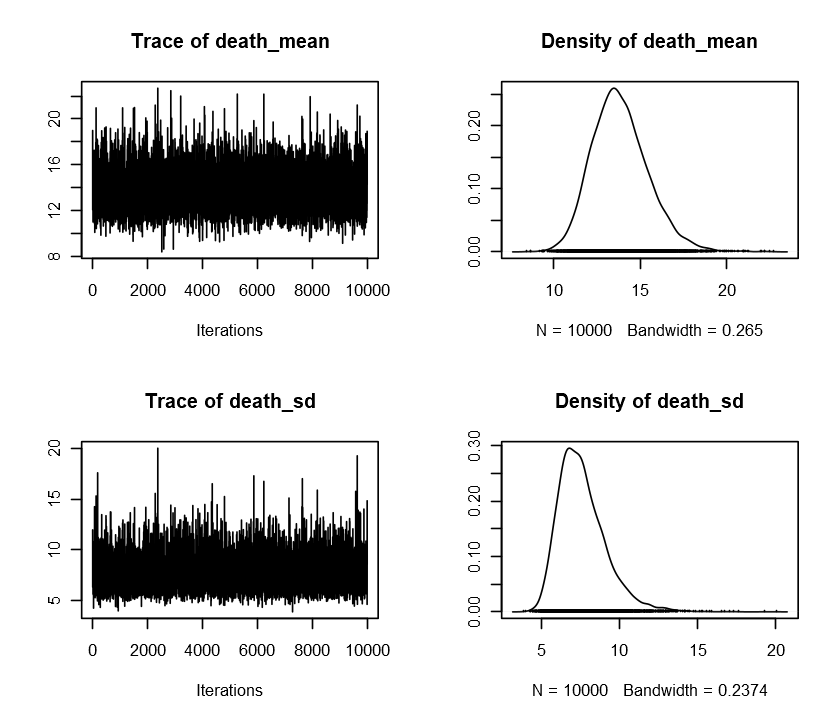
**Appendix Fig 1: Timeline of a nosocomial outbreak in Jos, Nigeria, in 1970, adapted from** [10]**.** Shaded grey areas indicate the exposure period, while shaded yellow areas span the time period from illness onset to death.



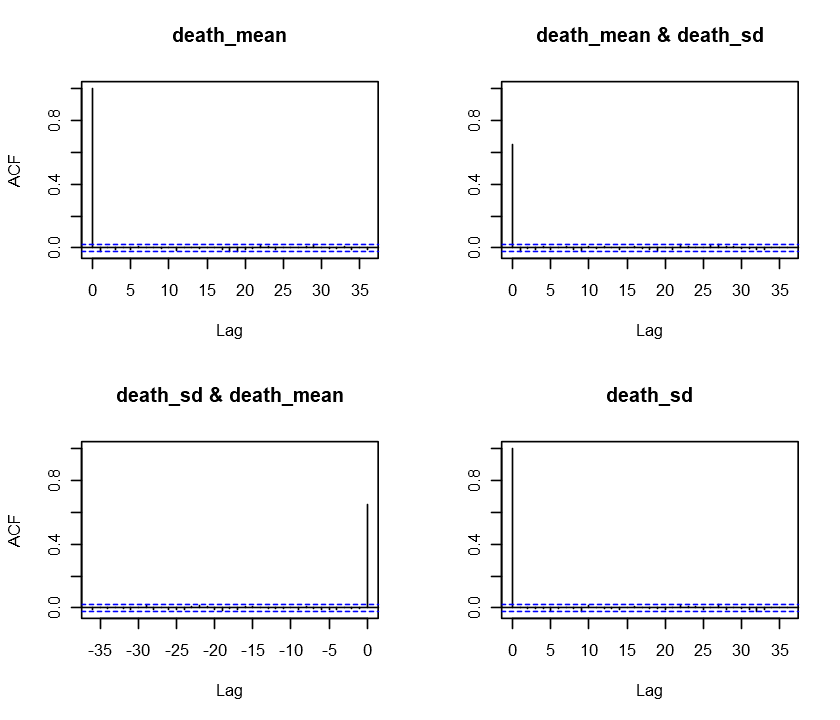
**Appendix Figure 2: Trace and density plots for the mean (“incubation\_mean”) and standard deviation (“incubation\_sd”) of the incubation period distribution.**



**Appendix Figure 3: Autocorrelation plots for pairs of mean (“incubation\_mean”) and standard deviation (“incubation\_sd”) values for the incubation period distribution.**



**Appendix Figure 4: Trace and density plots for the mean (“death\_mean”) and standard deviation (“death\_var”) of the distribution of the time period between illness onset and death.**



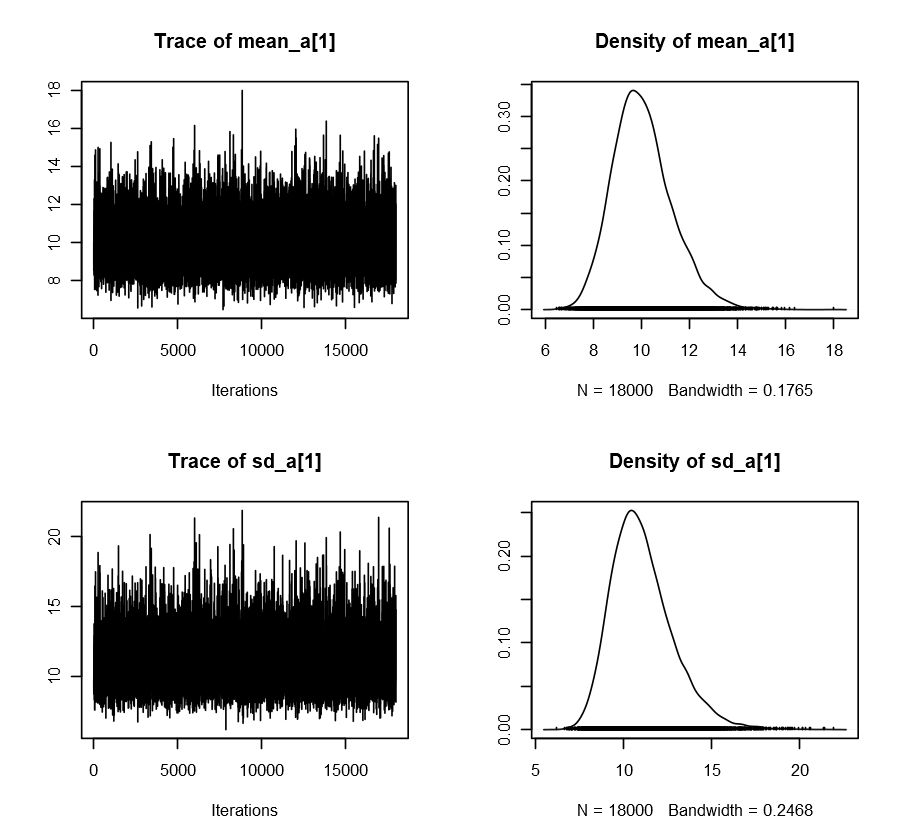
**Appendix Figure 5: Autocorrelation plots for pairs of mean (“death\_mean”) and standard deviation (“death\_sd”) values for the distribution of the time period between illness onset and death.**

****

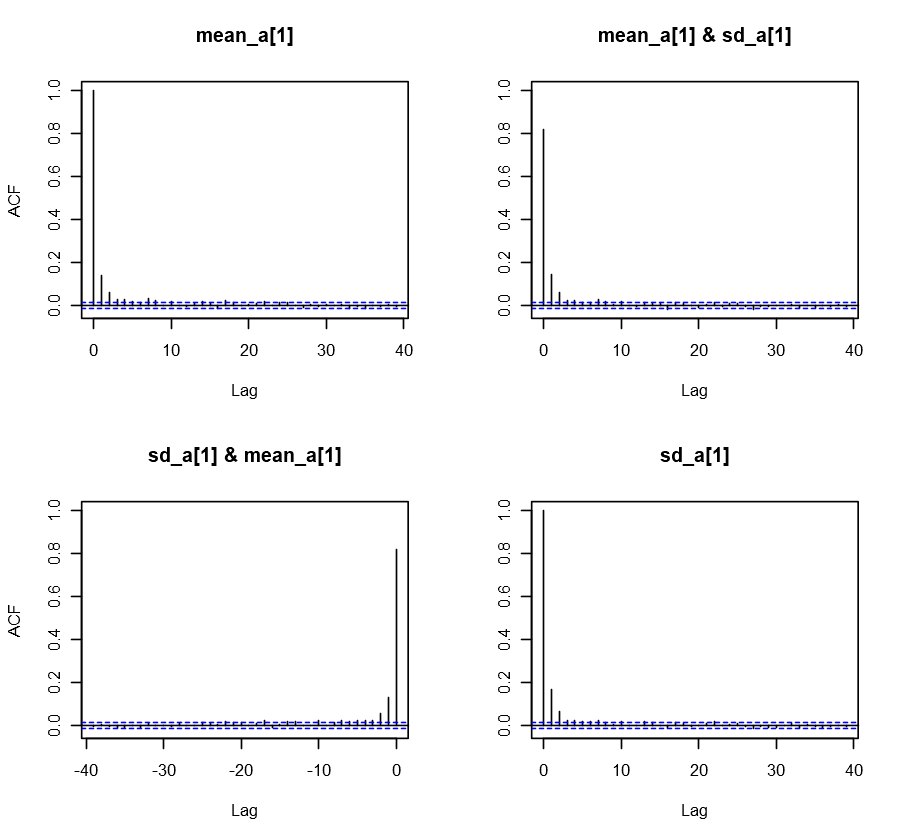
**Appendix Figure 6: Model fit to the observed data of new cases (*a*) and fatal cases (*b*) using maximum likelihood estimation.** Solid black line indicates the obtained average, whereas the shaded area shows the 95% profile-based confidence intervals. The fitted exposure rate as a function of calendar week is shown in the inset of (a).

****

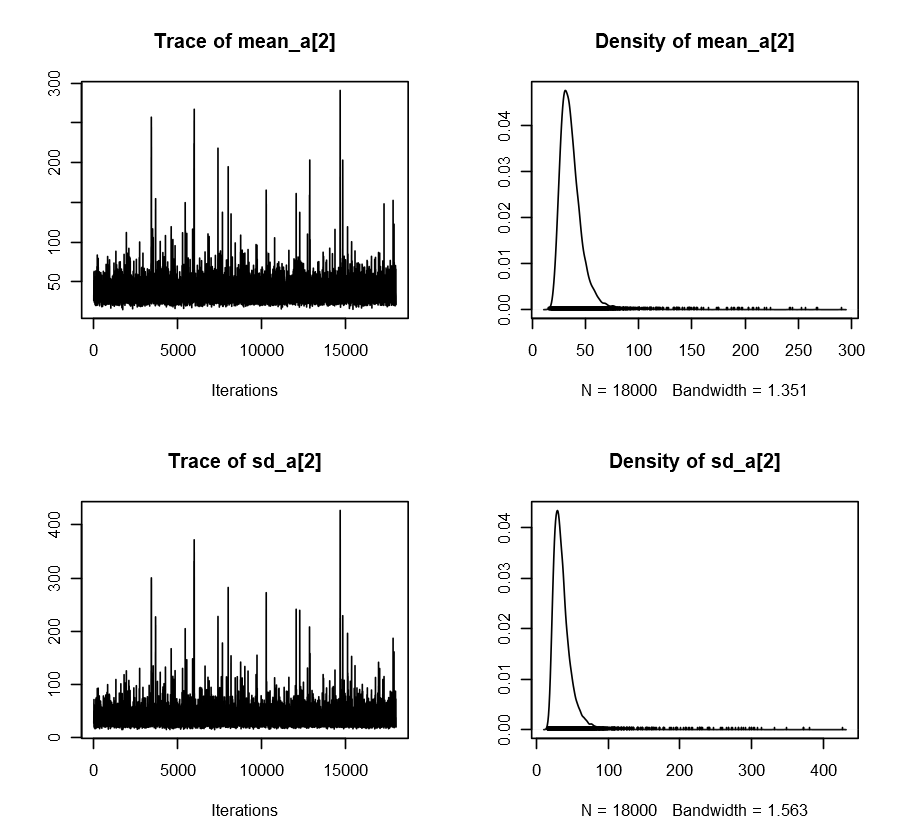
**Appendix Figure 7: Model fit to the observed data of new cases (*a*) and fatal cases (*b*) using maximum likelihood estimation for the alternative model with (single) constant exposure rate.** Solid black line indicates the obtained average, whereas the shaded area shows the 95% profile-based confidence intervals. The fitted exposure rate as a function of calendar week is shown in the inset of (a).



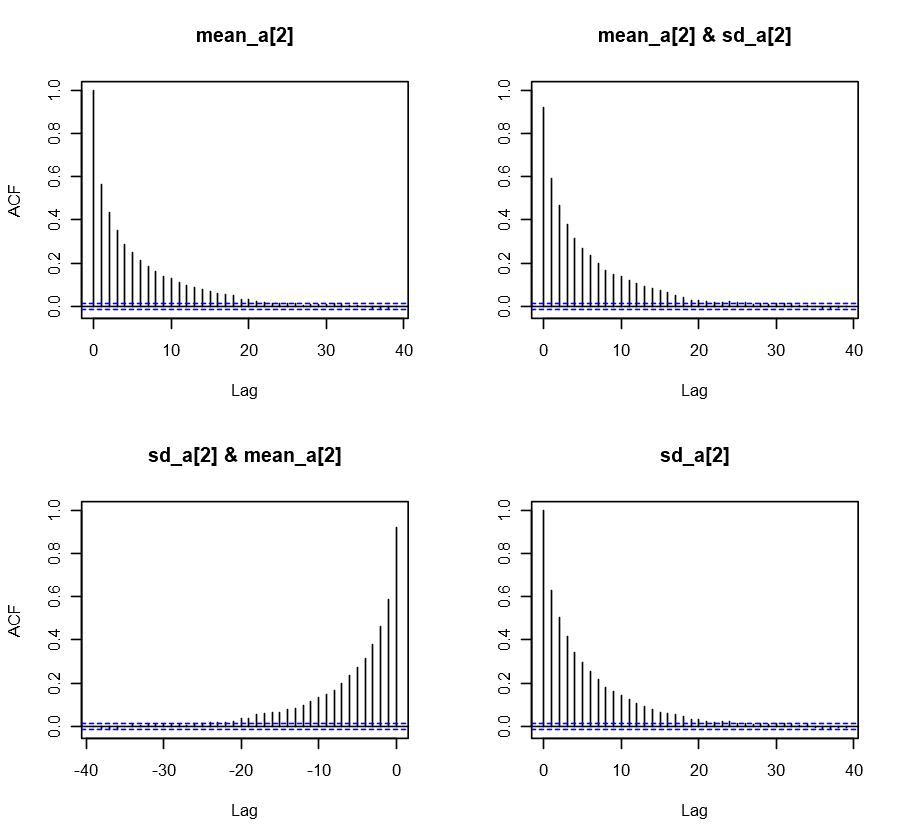
**Appendix Figure 8: Trace and density plots for and .**

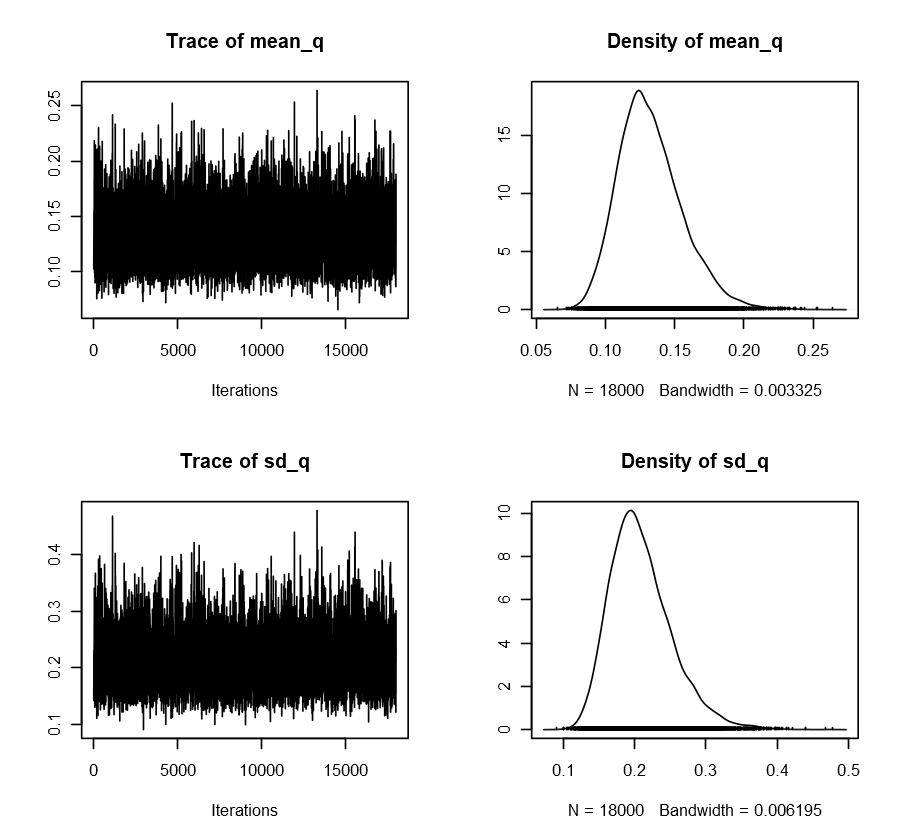


**Appendix Figure 9: Autocorrelation plots for the pair and .**

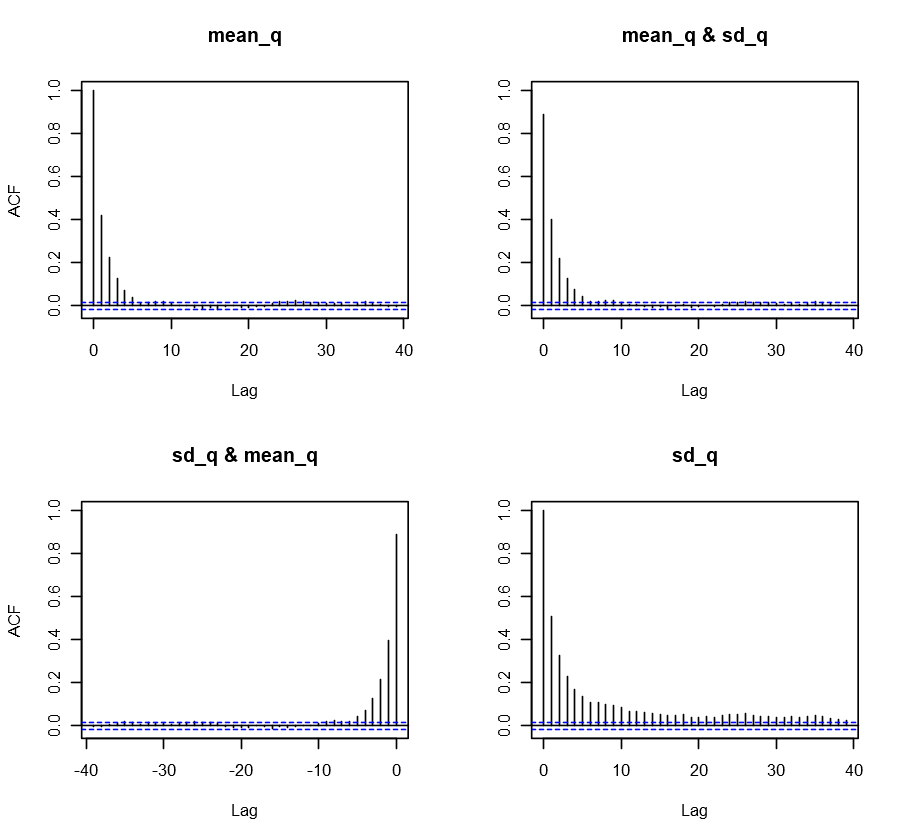


**Appendix Figure 10: Trace and density plots for and .**

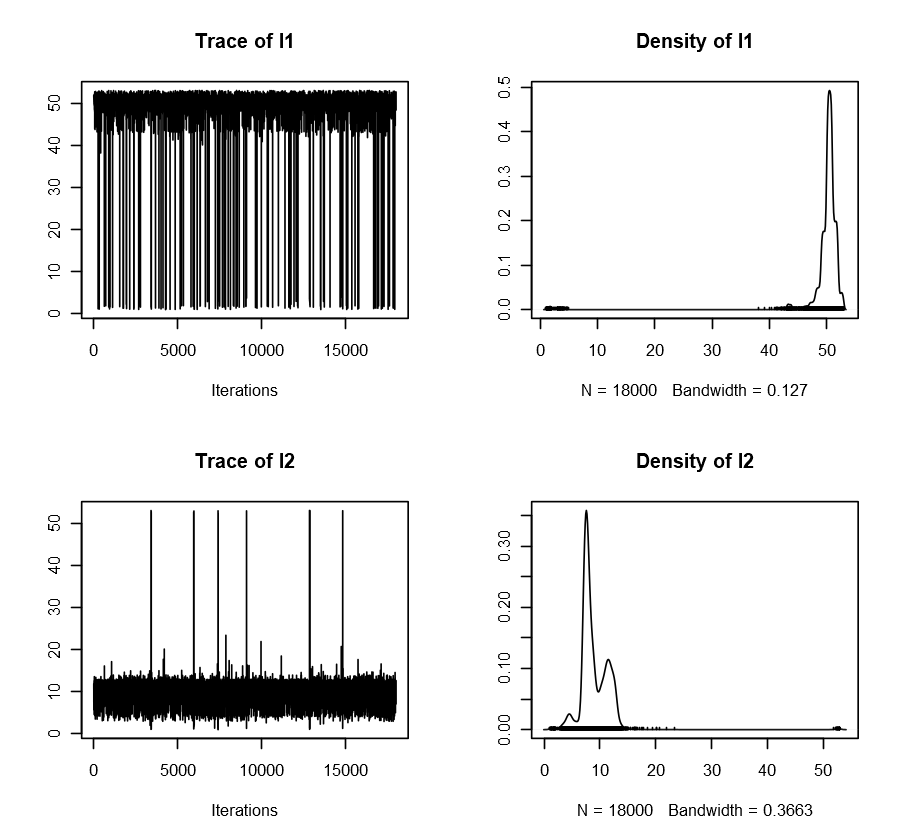
**Appendix Figure 11: Autocorrelation plots for the pair and .**

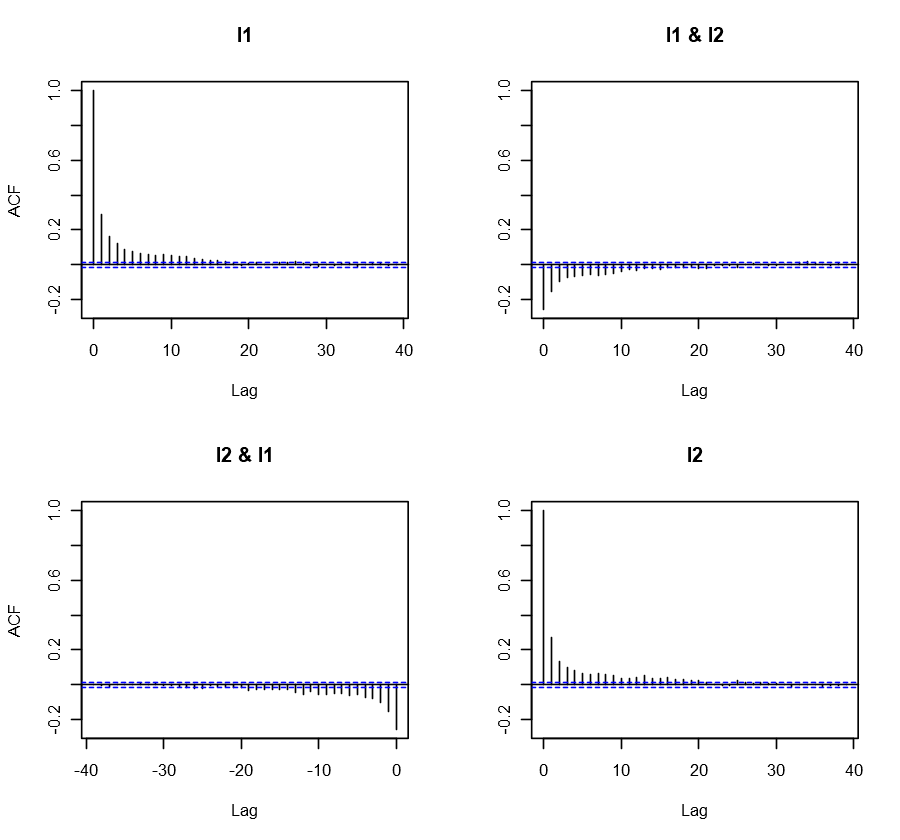


**Appendix Figure 12: Trace and density plots for and .**

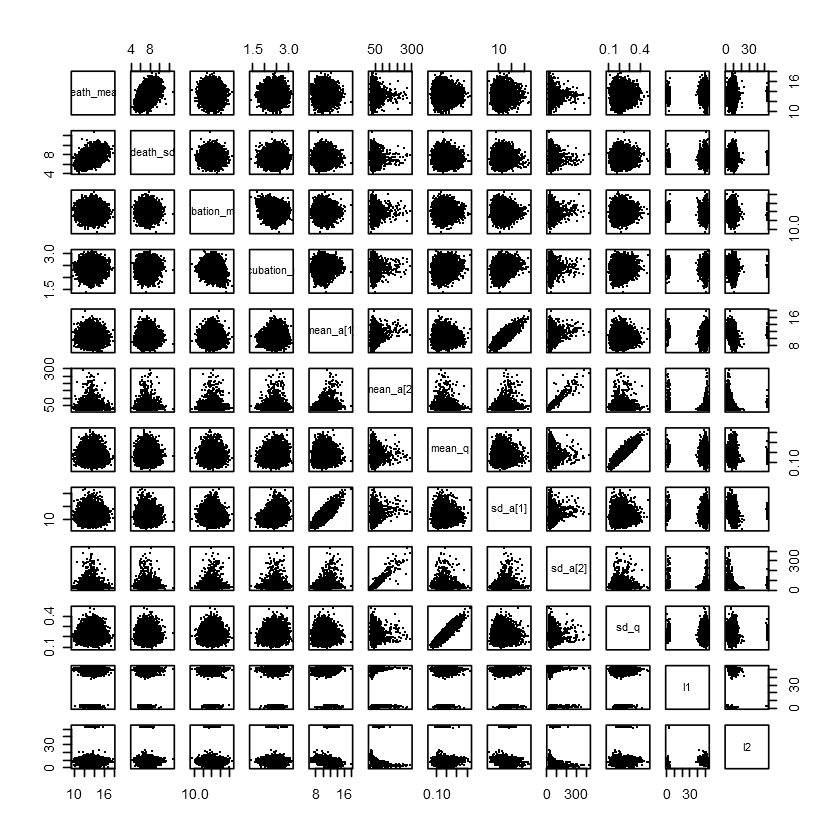


**Appendix Figure 13: Autocorrelation plots for the pair and .**

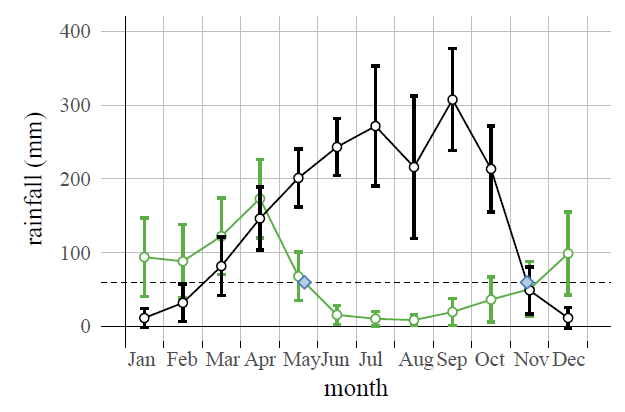
**Appendix Figure 14: Trace and density plots for and .**



**Appendix Figure 15: Autocorrelation plots for the pair and .**



**Appendix Figure 16: Inference of the model parameters using MCMC iterations.**

****

**Appendix Figure 17: Difference in rainfall patterns between Nigeria (black) and Tanzania (green).** The location used for Tanzania was S, E, which was the data collection site in the original study [11]. The closeness was restrained by gridded data point distribution in the global precipitation dataset [12]. Blue points indicate the starting points for the dry seasons.



****

**Appendix Figure 18: Numerical minimization of the value over two parameters and .**

**Appendix Code Snippet 1: Main R script including the NIMBLE module**

## Preamble

args = commandArgs(trailingOnly=TRUE)

set.seed(as.numeric(args[1]))

libraries = c("dplyr","magrittr","tidyr","readxl","nimble")

for(x in libraries) { library(x,character.only=TRUE,warn.conflicts=FALSE) }

'%&%' = function(x,y)paste0(x,y)

rho = 0.19

## Data for to describe Jos outbreak

#### excluding index case

onsetTimes = c(17, 19, 21, 21, 21, 22, 22, 23, 23, 23, 23, 23, 25, 26, 26, 26, 27, 31, 39, 41, 41, 42, 44)

deathTimes = c(33, 48, 31, 47, 30, 41, 37, 32, 31, 35, 54, 30, 31, 33, 30, 33, 48, 39, 51, 48, 55, 60, 62)

## Prior knowledge

exposureTimesLower = c(5, 9, 5, 9, 5, 5, 5, 11, 5, 5, 13, 11, 12, 5, 11, 11, 5, 5, 12, 25, 5, 25, 25)

exposureTimesUpper = c(15, 18, 18, 15, 18, 18, 18, 15, 18, 18, 14, 15, 18, 18, 15, 15, 18, 18, 18, 31, 18, 31, 31)

## Dataset of human incidence stored in Nigeria\_raw.xlsx

yearMin = 2016

data = read\_excel("../../data/Nigeria\_raw.xlsx", sheet = "Incidence") %>%

select(-one\_of("Timeseries","Imputation","File in the repo"),-contains("URL")) %>%

filter(Year>=yearMin) %>%

group\_by(Year) %>%

mutate(Incidence\_Reported = if\_else(Week==1,Reported,Reported-lead(Reported)),

Incidence\_Deaths = if\_else(Week==1,Deaths,Deaths-lead(Deaths))

) %>%

ungroup

data %>% tail

data %>% select(Year,Week,starts\_with("Incidence")) -> Df

data.frame(Year=yearMin-1,Week=1:52,Incidence\_Reported=NA,Incidence\_Deaths=NA) %>%

rbind(Df %>% arrange(Year,Week)) %>%

rowwise %>%

mutate(Incidence\_Reported\_NA=ifelse(is.na(Incidence\_Reported),rpois(1,30),NA),

Incidence\_Deaths\_NA=ifelse(is.na(Incidence\_Deaths),rpois(1,1),NA)) %>%

ungroup -> Df

(K = nrow(Df))

# Convolutions = calculation of initial values for MCMC simulations

# that are used as initial values for the followed inference

incubation\_shape = 8.038

incubation\_rate = 1/0.2278

death\_shape = 3.3012

death\_rate = 1/0.5968

incubation\_probability = pgamma(1:K,shape=incubation\_shape,rate=incubation\_rate)-pgamma(1:K-1,shape=incubation\_shape,rate=incubation\_rate)

timeFromOnsetToDeath\_probability = pgamma(1:K,shape=death\_shape,rate=death\_rate)-pgamma(1:K-1,shape=death\_shape,rate=death\_rate)

# time from Exposure event to Death is the convolution of two latter probabilities

timeFromExposureToDeath\_probability = c(0)

for (x in 2:K) {

timeFromExposureToDeath\_probability = c(timeFromExposureToDeath\_probability,

sum(incubation\_probability[1:(x-1)]\*timeFromOnsetToDeath\_probability[(x-1):1]))

}

### Machinery for Nimble

# Convolution functions

nimConvolutionWithCFR = nimbleFunction(

run = function(a = double(1), b = double(1), q = double(1)) {

L <- dim(a)[1]

res1 <- numeric(L)

for(k in 1:L) {

res1[k] <- q[k]\*a[k]

}

ans <- inprod(res1[L+1-1:L],b)

return(ans)

returnType(double(0))

}

)

nimConvolution = nimbleFunction(

run = function(a = double(1), b = double(1)) {

L <- dim(a)[1]

ans <- inprod(a[L+1-1:L],b)

return(ans)

returnType(double(0))

}

)

## Shift in l1 and l2 used to avoid edge effects

shift = 26

nCases = length(onsetTimes)

## Main script

nimData = list(# incubation and period to death

onsetTime = onsetTimes,

deathTime = deathTimes,

# incidence

infected = Df$Incidence\_Reported,

dead = Df$Incidence\_Deaths,

# for contraint

one = 1

)

# the following values are used for priors

incubation\_mean\_median = 13.16830

incubation\_mean\_lower = 10.99341

incubation\_mean\_upper = 15.52661

incubation\_sd\_median = 5.257898

incubation\_sd\_lower = 3.530271

incubation\_sd\_upper = 7.312072

death\_mean\_median = 14.21680

death\_mean\_lower = 10.83839

death\_mean\_upper = 17.70390

death\_sd\_median = 8.033101

death\_sd\_lower = 5.157413

death\_sd\_upper = 11.476860

nimConsts = list(nCases = length(onsetTimes),

incubation\_uncertainty = 0.5,

death\_uncertainty = 0.5,

exposure\_mu = .5\*(exposureTimesLower+exposureTimesUpper),

exposure\_sd = (exposureTimesUpper-exposureTimesLower)/1.96/2,

r = rho,

K = K,

week = (Df$Week-shift-1)%%52+1,

incubation\_mean\_mean = (incubation\_mean\_upper+incubation\_mean\_lower)/2,

incubation\_mean\_sd = (incubation\_mean\_upper-incubation\_mean\_lower)/1.96/2,

incubation\_sd\_mean = (incubation\_sd\_upper+incubation\_sd\_lower)/2,

incubation\_sd\_sd = (incubation\_sd\_upper-incubation\_sd\_lower)/1.96/2,

death\_mean\_mean = (death\_mean\_upper+death\_mean\_lower)/2,

death\_mean\_sd = (death\_mean\_upper-death\_mean\_lower)/1.96/2,

death\_sd\_mean = (death\_sd\_upper+death\_sd\_lower)/2,

death\_sd\_sd = (death\_sd\_upper-death\_sd\_lower)/1.96/2

)

nimInits = function(){

lambdaIncidence0 = rexp(K,1/3); CFR0 = runif(K,0.02,0.12);

list(# incubation and period to death

incubation\_mean = runif(1,incubation\_mean\_lower,incubation\_mean\_upper),

incubation\_sd = runif(1,incubation\_sd\_lower,incubation\_sd\_upper),

death\_mean = runif(1,death\_mean\_lower,death\_mean\_upper),

death\_sd = runif(1,death\_sd\_lower,death\_sd\_upper),

onsetExpectedTime = onsetTimes,

deathExpectedTime = deathTimes,

exposureTime = .5\*(exposureTimesLower+exposureTimesUpper),

incubationTime = onsetTimes-.5\*(exposureTimesLower+exposureTimesUpper),

timeToDeath = deathTimes-onsetTimes,

# results

incubationPeriod = incubation\_probability,

timeFromOnsetToDeath = timeFromOnsetToDeath\_probability,

timeFromExposureToDeath = timeFromExposureToDeath\_probability,

# incidence

l1 = (runif(1,44,52)-shift-1)%%52+1,

l2 = (runif(1,5,12)-shift-1)%%52+1,

infected = Df$Incidence\_Reported\_NA,

dead = Df$Incidence\_Deaths\_NA,

lambdaIncidence = (1-CFR0)\*lambdaIncidence0,

lambdaDeath = CFR0\*lambdaIncidence0,

lambdaReported = lambdaIncidence0,

pDeath = CFR0,

exposure = lambdaIncidence0,

CFR = CFR0,

mean\_a = c(runif(1,5,15),runif(1,.5,3)),

sd\_a = runif(2,0.1,1),

mean\_q = runif(1,0.02,0.12),

sd\_q = runif(1,0.1,1))}

nimCode = nimbleCode({

### incubation period and period from illness onset to death

incubation\_mean ~ dnorm(mean=incubation\_mean\_mean, sd=incubation\_mean\_sd)

incubation\_sd ~ dnorm(mean=incubation\_sd\_mean, sd=incubation\_sd\_sd)

death\_mean ~ dnorm(mean=death\_mean\_mean, sd=death\_mean\_sd)

death\_sd ~ dnorm(mean=death\_sd\_mean, sd=death\_sd\_sd)

for (k in 1:nCases) {

exposureTime[k] ~ dnorm(mean=exposure\_mu[k],sd=exposure\_sd[k])

incubationTime[k] ~ dgamma(mean=incubation\_mean, sd=incubation\_sd)

timeToDeath[k] ~ dgamma(mean=death\_mean, sd=death\_sd)

onsetExpectedTime[k] <- exposureTime[k] + incubationTime[k]

deathExpectedTime[k] <- onsetExpectedTime[k] + timeToDeath[k]

onsetTime[k] ~ dnorm(onsetExpectedTime[k], sd=incubation\_uncertainty)

deathTime[k] ~ dnorm(deathExpectedTime[k], sd=death\_uncertainty)

}

### epidemiological model

incubation\_shape <- incubation\_mean^2/incubation\_sd^2

incubation\_rate <- incubation\_mean/incubation\_sd^2\*7.0 #from days to weeks

death\_shape <- death\_mean^2/death\_sd^2

death\_rate <- death\_mean/death\_sd^2\*7.0 #from days to weeks

for (k in 1:K) {

incubationPeriod[k] <- pgamma(k,shape=incubation\_shape,rate=incubation\_rate)-pgamma(k-1,shape=incubation\_shape,rate=incubation\_rate)

timeFromOnsetToDeath[k] <- pgamma(k,shape=death\_shape,rate=death\_rate)-pgamma(k-1,shape=death\_shape,rate=death\_rate)

}

timeFromExposureToDeath[1] <- 0

for (k in 1:(K-1)) {

timeFromExposureToDeath[k+1] <- nimConvolution(incubationPeriod[1:k],timeFromOnsetToDeath[1:k])

}

### exposure

for (k in 1:K) {

mean\_a\_realized[k] <- mean\_a[1]+(mean\_a[2]-mean\_a[1])\*equals(step(week[k]-l1),step(l2-week[k]))

sd\_a\_realized[k] <- sd\_a[1]+(sd\_a[2]-sd\_a[1])\*equals(step(week[k]-l1),step(l2-week[k]))

exposure[k] ~ dgamma(mean=mean\_a\_realized[k],sd=sd\_a\_realized[k])

CFR[k] ~ dgamma(mean=mean\_q,sd=sd\_q)

}

for (k in 52:(K-1)) {

lambdaIncidence[k+1] <- nimConvolutionWithCFR(exposure[1:k],incubationPeriod[1:k],1-CFR[1:k])/(1-r)

lambdaDeath[k+1] <- nimConvolutionWithCFR(exposure[1:k],timeFromExposureToDeath[1:k],CFR[1:k])/(1-r)

lambdaReported[k+1] <- lambdaDeath[k+1]+lambdaIncidence[k+1]

pDeath[k+1] <- lambdaDeath[k+1]/lambdaReported[k+1]

}

for (k in 53:K) {

infected[k] ~ dpois(lambdaReported[k])

dead[k] ~ dbin(pDeath[k],infected[k])

}

## Priors

for(i in 1:2) {

mean\_a[i] ~ dhalfflat()

sd\_a[i] ~ dhalfflat()

}

mean\_q ~ dhalfflat()

sd\_q ~ dhalfflat()

l1 ~ dunif(0,53)

l2 ~ dunif(0,53)

one ~ dconstraint(l2>l1-3)

})

nimModel = nimbleModel(nimCode,

constants = nimConsts,

data = nimData,

inits = nimInits())

## Checking that all variables are properly initialized

nimModel$initializeInfo()

nimConf = configureMCMC(nimModel, thin = 100, setSeed=TRUE)

nimConf$addMonitors(c("incubationTime","timeToDeath","lambdaIncidence","lambdaDeath","pDeath","exposure","CFR","infected","incubationPeriod","timeFromOnsetToDeath","timeFromExposureToDeath"))

## Model compilation

nimMCMC = buildMCMC()

compiledModel = compileNimble(nimModel, nimMCMC, resetFunctions = TRUE, showCompilerOutput = TRUE)

Niter = 1e5

Nburn = 2e4

compiledModel$nimMCMC$run(niter=Niter+Nburn, nburnin = Nburn)

compiledModel$nimMCMC$mvSamples %>% as.matrix %>% as.data.frame -> nimSamples

saveRDS(nimSamples, file = paste0("nimSamples-",args[1],".rds"))

compiledModel$nimMCMC$getTimes() %>% { sum(.)/60 }

**Appendix Code Snippet 2: An example of a bash script applied to Appendix Code Snippet 1 “MCMC.r”**

#!/bin/bash

for replicate in {1..25}

do

R CMD BATCH --no-save --no-restore '--args '"${replicate}" MCMC.r MCMC-final-${replicate}.Rout &

done