

Sensitivity analysis of deterministic models through Latin hypercube sampling: A model for the spread of Ebola virus disease*

John M. Drake & Pejman Rohani

A model for the transmission of Ebola virus disease

Ebola virus is an emerging pathogen of humans and other non-human primates (Alexander *et al.*, 2015). Outbreaks are believed to originate from contact with an infected reservoir (possibly a bat) or consumption of infected bushmeat. Outbreaks in human settlements are facilitated by nosocomial transmission when sufficient infection control procedures are not in place, transmission by care-givers in the community, and traditional burial practices that may expose a large number of persons simultaneously. Approximately 30 human outbreaks occurred prior to the West Africa epidemic of 2014-2015 (Alexander *et al.*, 2015).

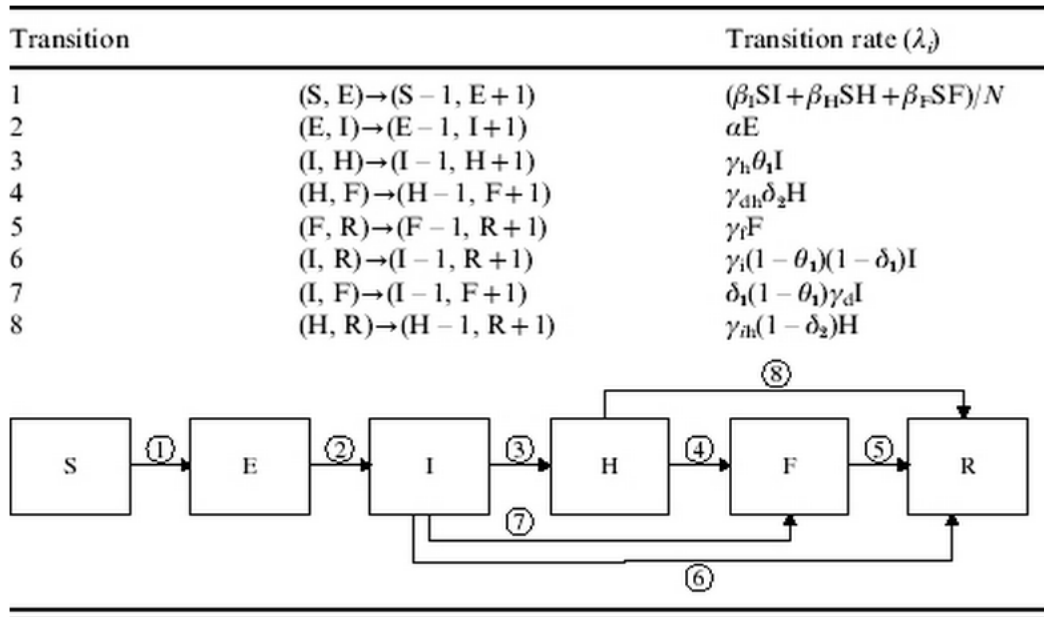


Figure 1: Compartmental model for Ebola virus transmission. Adapted from Table 2 in Legrand *et al.* (2007).

*Licensed under the Creative Commons attribution-noncommercial license, <http://creativecommons.org/licenses/by-nc/3.0/>. Please share and remix noncommercially, mentioning its origin.

Legrand *et al.* (2007) used data from two of these outbreaks – Democratic Republic of Congo (1995) and Uganda (2000) – to fit a model reflecting these multiple sites of transmission. The model they proposed (Fig. 1) has six compartments: susceptible persons (S), exposed (incubating) persons (E), infectious persons (I), hospitalized patients (H), deceased patients that may transmit during funeral proceedings (F), and persons removed from the population (R). Parameters include transmission coefficient in the community (β_I), transmission coefficient in the hospital (β_H), transmission coefficient during funerals (β_F), θ_1 is estimated to ensure that the observed fraction of infectious cases are hospitalized, δ_1 and δ_2 are estimated so that the overall case fatality rate is as observed. Average incubation period is α^{-1} , average time from onset of symptoms to hospitalization is γ_h^{-1} , average time from hospitalization to death is γ_{dh}^{-1} , mean infectious period for survivors is γ_i^{-1} , mean time from hospitalization to recovery is γ_{ih}^{-1} , and mean time from death to burial is γ_f^{-1} . The transition rate from hospitalization to recovery is further related to other parameters through the relation $\gamma_{ih} = \frac{\gamma_i \gamma_h}{\gamma_h - \gamma_i}$ while the transition rate from hospitalization to funeral is $\gamma_{dh} = \frac{\gamma_d \gamma_h}{\gamma_h - \gamma_d}$.

Numerical solution

This flow diagram can be expressed as a system of differential equations. Our objectives in this exercise are:

1. Numerically solve the differential equation
2. Use latin hypercube sampling to explore the robustness of model results to uncertainty in a 9-dimensional parameter space

Exercise 1. In R, write a function to return the rates of change for this model. Use `deSolve` to numerically solve at the values obtained by Legrand *et al.* (2007) for Uganda: $\beta_I = 2.532$, $\beta_H = 0.012$, $\beta_F = 0.462$, $\alpha = 1/12$, $\gamma_h = 1/4.2$, $\gamma_d = 1/8$, $\gamma_i = 1/10$, $\gamma_f = 1/2$, $\gamma_d = 1/8$, $\theta_1 = 0.65$, $\delta_1 = 0.47$, $\delta_2 = 0.42$. Confusingly, β_x parameters are in units of w^{-1} while α and γ_x parameters are in units of d^{-1} . Be sure to standardize to a common timescale. Additionally, assume the population size is 470,000 and that the outbreak was initiated by nine cases.

Exercise 2. Plot the six state variables for the first 52 weeks of the outbreak.

```
> legrand <- function(t, x, params){
+
+   # states
+   S <- x[1]
+   E <- x[2]
+   I <- x[3]
+   H <- x[4]
+   F <- x[5]
+   R <- x[6]
+   N <- S+E+I+H+F+R
+
+   # calculated parameters
+   gammaih = params$gammai*params$gammah/(params$gammah - params$gammai)
+   gammadh = params$gammad*params$gammah/(params$gammah - params$gammad)
+   # rates
+   r1 <- (params$betaI*S*I + params$betaH*S*H + params$betaF*S*F)/N
+   r2 <- params$alpha*E
+   r3 <- params$gammah*params$theta1*I
```

```

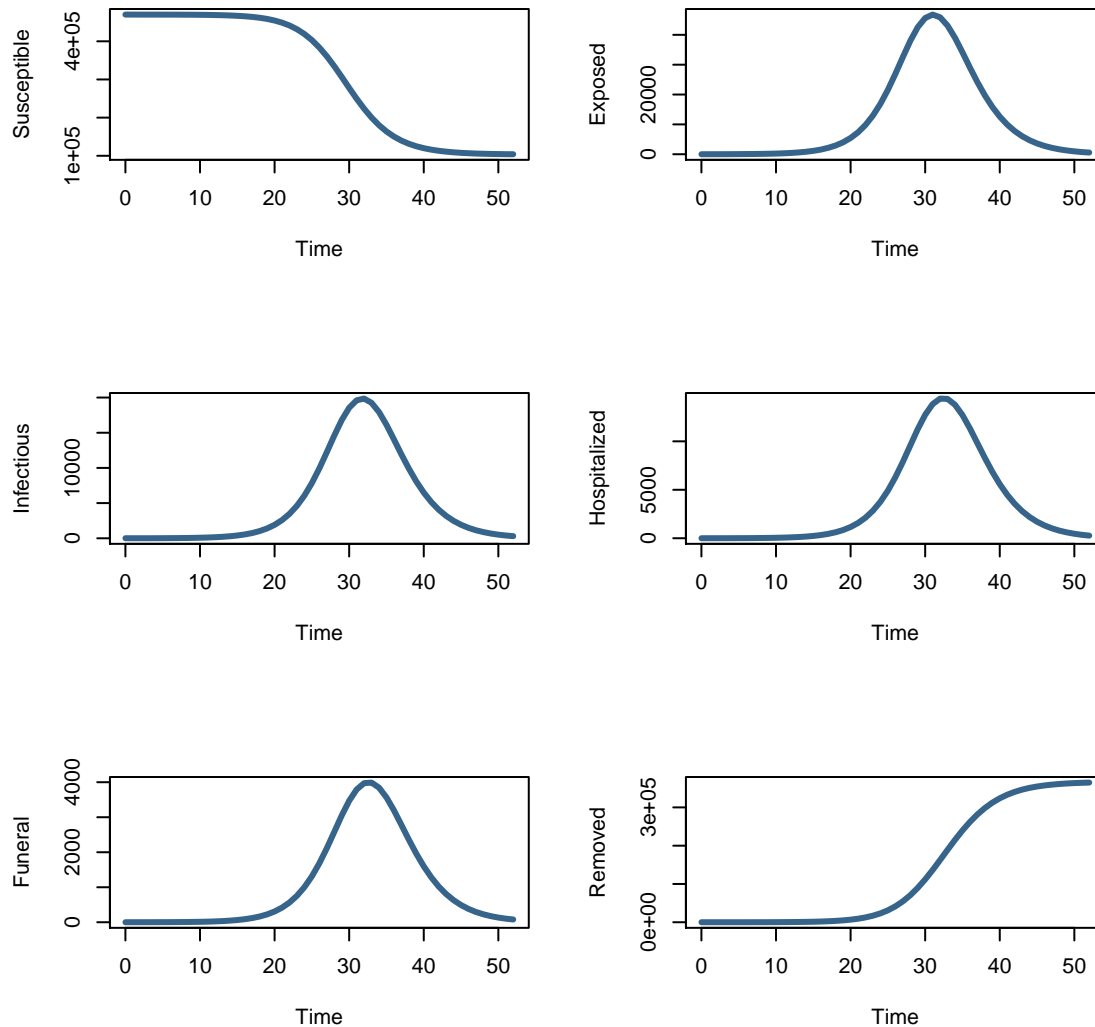
+ r4 <- gammadh*params$delta2*H
+ r5 <- params$gammaf*F
+ r6 <- params$gammai*(1-params$theta1)*(1-params$delta1)*I
+ r7 <- params$delta1*(1-params$theta1)*params$gammad*I
+ r8 <- gammaih*(1-params$delta2)*H
+
+ # derivatives
+ dS <- -r1
+ dE <- r1 - r2
+ dI <- r2 - r3 - r7 - r6
+ dH <- r3 - r4 - r8
+ dF <- r4 + r7 - r5
+ dR <- r5 + r6 + r8
+
+ #output
+ out <- list(c(dS, dE, dI, dH, dF, dR))
+ }

> require(deSolve)
> times <- seq(0, 52, by=1) #solve for 52 weeks
> params <- list(betaI=2.532,
+               betaH=0.012,
+               betaF=0.462,
+               alpha=7/12,
+               gammah=7/4.2,
+               gammai=7/10,
+               gammaf=7/2,
+               gammad = 7/8,
+               theta1=0.65,
+               delta1=0.47,
+               delta2=0.42)
> pop.size <- 470000
> I0 <- 9
> xstart <- c(S=pop.size-I0, E=0, I=I0 , H=0, F=0, R=0)
> out <- as.data.frame(ode(xstart, times, legrand, params))

> plot.legrand <- function(out){
+   par(mfrow=c(3,2))
+   plot(out$time, out$S, xlab='Time', ylab='Susceptible', type='l', col='steelblue4', lwd=3)
+   plot(out$time, out$E, xlab='Time', ylab='Exposed', type='l', col='steelblue4', lwd=3)
+   plot(out$time, out$I, xlab='Time', ylab='Infectious', type='l', col='steelblue4', lwd=3)
+   plot(out$time, out$H, xlab='Time', ylab='Hospitalized', type='l', col='steelblue4', lwd=3)
+   plot(out$time, out$F, xlab='Time', ylab='Funeral', type='l', col='steelblue4', lwd=3)
+   plot(out$time, out$R, xlab='Time', ylab='Removed', type='l', col='steelblue4', lwd=3)
+ }

> plot.legrand(out)

```



This exercise will investigate the effectiveness of early interventions at reducing the total epidemic size.

Exercise 3. Modify your model to allow interventions at time T . Legrand *et al.* assumed that interventions were completely efficient after T such that, after interventions (i) no transmission occurred at hospitals or during burial, and (ii) transmission in the community was decreased by a factor $(1 - z)$. In Uganda, interventions are assumed to have been initiated on the date that Ebola virus disease was confirmed (15 October 2015). Although Legrand *et al.* (2007) do not state this, a comparison of Figure 1 in their paper and Figure 2 in Oyok *et al.* (2001) shows that Legrand *et al.* are dating the outbreak from 10 September 2015.

```
> # model of Legrand et al (2007) with interventions
> legrand2 <- function(t, x, params){
+
+   # states
+   S <- x[1]
+   E <- x[2]
```

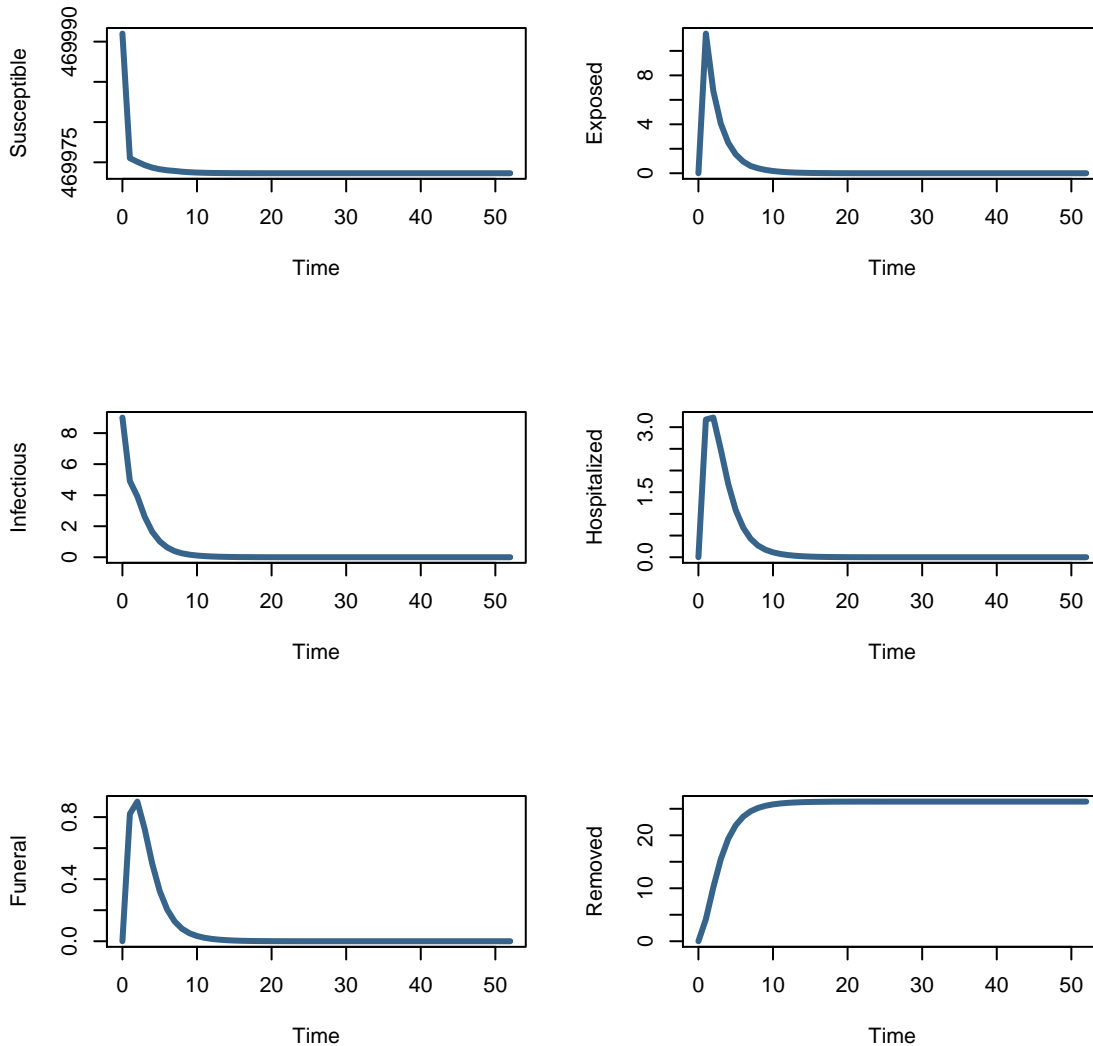
```

+ I <- x[3]
+ H <- x[4]
+ F <- x[5]
+ R <- x[6]
+ N <- S+E+I+H+F+R
+
+ # calculated parameters
+ gammaih = params$gammai*params$gammah/(params$gammah - params$gammai)
+ gammadh = params$gammad*params$gammah/(params$gammah - params$gammad)
+
+ # rates
+ r1 <- (params$betaI*S*I*(t<T) + params$betaI*S*I*(1-params$z)*(t>=params$T) + params$betaH*S*H*(t<T) +
+ r2 <- params$alpha*E
+ r3 <- params$gammah*params$theta1*I
+ r4 <- gammadh*params$delta2*H
+ r5 <- params$gammaf*F
+ r6 <- params$gammai*(1-params$theta1)*(1-params$delta1)*I
+ r7 <- params$delta1*(1-params$theta1)*params$gammad*I
+ r8 <- gammaih*(1-params$delta2)*H
+
+ # derivatives
+ dS <- -r1
+ dE <- r1 - r2
+ dI <- r2 - r3 - r7 - r6
+ dH <- r3 - r4 - r8
+ dF <- r4 + r7 - r5
+ dR <- r5 + r6 + r8
+
+ #output
+ out <- list(c(dS, dE, dI, dH, dF, dR))
+ }

> params2 <- list(betaI=2.532,
+ betaH=0.012,
+ betaF=0.462,
+ alpha=7/12,
+ gammah=7/4.2,
+ gammai=7/10,
+ gammaf=7/2,
+ gammad = 7/8,
+ theta1=0.65,
+ delta1=0.47,
+ delta2=0.42,
+ T=7,
+ z=0.88)
> pop.size <- 470000
> I0 <- 9
> xstart <- c(S=pop.size-I0, E=0, I=I0, H=0, F=0, R=0)
> out2 <- as.data.frame(ode(xstart, times, legrand2, params2))

> plot.legrand(out2)

```



Some of these parameters are rather poorly estimated and there is an argument to be made that the transmission rates are at best weakly identifiable. How much does this uncertainty influence our conclusions? We can answer this question using latin hypercube sampling. The first step is to determine how best to summarize each parameterization. A natural suggestion is the total epidemic size, which may be extracted from the R class in the final row of the ode output using the function `tail`.

```
> get.size <- function(out) tail(out$R,1)
```

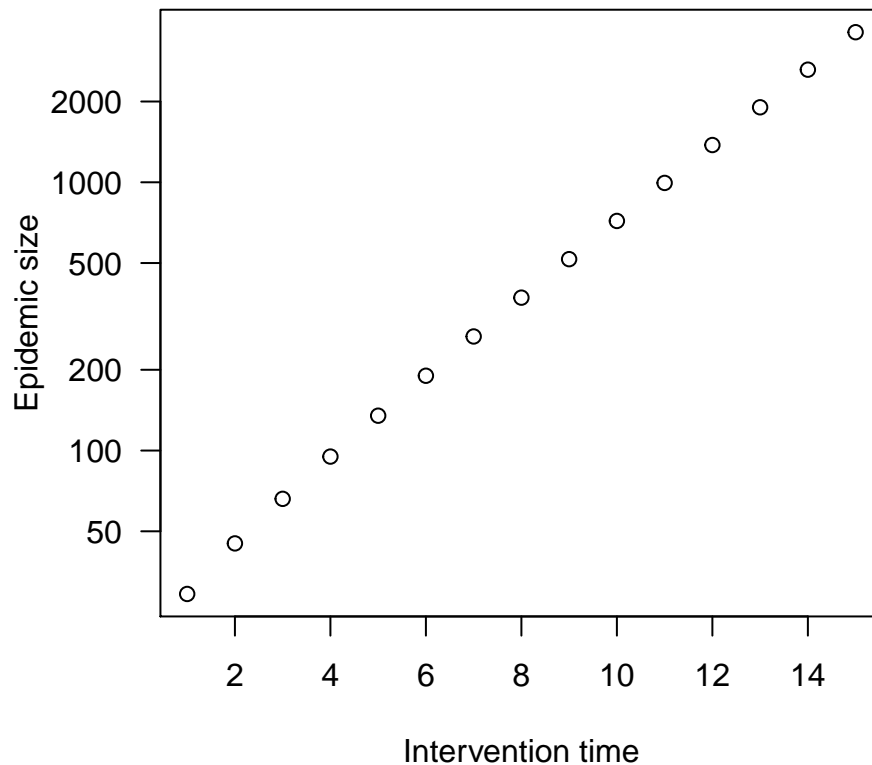
First, as a baseline, we use T (the time of intervention) as a control parameter and investigate the effect of T on total epidemic size. Evidently, the total epidemic size increases almost exponentially with the time to intervention. (Note the log scale on the y-axis.)

```
> output0 <- rep(NA, 15)
> intervention.times <- seq(1,15)
```

```

> for(T in intervention.times){
+   params2$T <- T
+   out3 <- as.data.frame(ode(xstart, times, legrand2, params2))
+   output0[T] <- get.size(out3)
+ }
> plot(intervention.times, output0, log='y', xlab='Intervention time', ylab='Epidemic size', las=1)

```



Latin hypercube sampling

Now we seek to use latin hypercube sampling to explore the 12-dimensional parameter space. First, use the `lhs` package to simulate a large number (say, 1000) points in the 12-dimensional unit cube.

```

> require(lhs)           #add the lhs library
> h <- 1000             #choose number of points
> set.seed(6242015)
> lhs<-maximinLHS(h,12) #simulate

```

To map these points in the unit cube to our parameters, we need minimum and maximum values for each.

```

> betaI.min <- 1
> betaI.max <- 4
> betaH.min <- 0.01
> betaH.max <- 0.5
> betaF.min <- 0.1
> betaF.max <- 4
> alpha.min <- 7/21
> alpha.max <- 7/2
> gammah.min <- 7/2
> gammah.max <- 7/1
> gammai.min <- 7/21
> gammai.max <- 7/2
> gammaf.min <- 7/7
> gammaf.max <- 7/1
> gammad.min <- 7/21
> gammad.max <- 7/2
> theta1.min <- 0
> theta1.max <- 1
> delta1.min <- 0
> delta1.max <- 1
> delta2.min <- 0
> delta2.max <- 1
> z.min <- 0
> z.max <- 1

```

Now we can generate a “parameter set” by rescaling our simulated latin hypercube sample.

```

> params.set <- cbind(
+   betaI = lhs[,1]*(betaI.max-betaI.min)+betaI.min,
+   betaH = lhs[,2]*(betaH.max-betaH.min)+betaH.min,
+   betaF = lhs[,3]*(betaF.max-betaF.min)+betaF.min,
+   alpha = lhs[,4]*(alpha.max-alpha.min)+alpha.min,
+   gammah = lhs[,5]*(gammah.max-gammah.min)+gammah.min,
+   gammai = lhs[,6]*(gammai.max-gammai.min)+gammai.min,
+   gammaf = lhs[,7]*(gammaf.max-gammaf.min)+gammaf.min,
+   gammad = lhs[,8]*(gammad.max-gammad.min)+gammad.min,
+   theta1 = lhs[,9]*(theta1.max-theta1.min)+theta1.min,
+   delta1 = lhs[,10]*(delta1.max-delta1.min)+delta1.min,
+   delta2 = lhs[,11]*(delta2.max-delta2.min)+delta2.min,
+   z = lhs[,12]*(z.max-z.min)+z.min)

```

These are all the simulated parameters we need to explore how these parameters affect the total epidemic size. To facilitate later plotting, we decide how many different levels of T to consider.

```

> levels <- 15

```

Even though we have simulated 1000 points, to speed up computing we will use only a fraction of these for this demonstration.

```

> h2 <-250

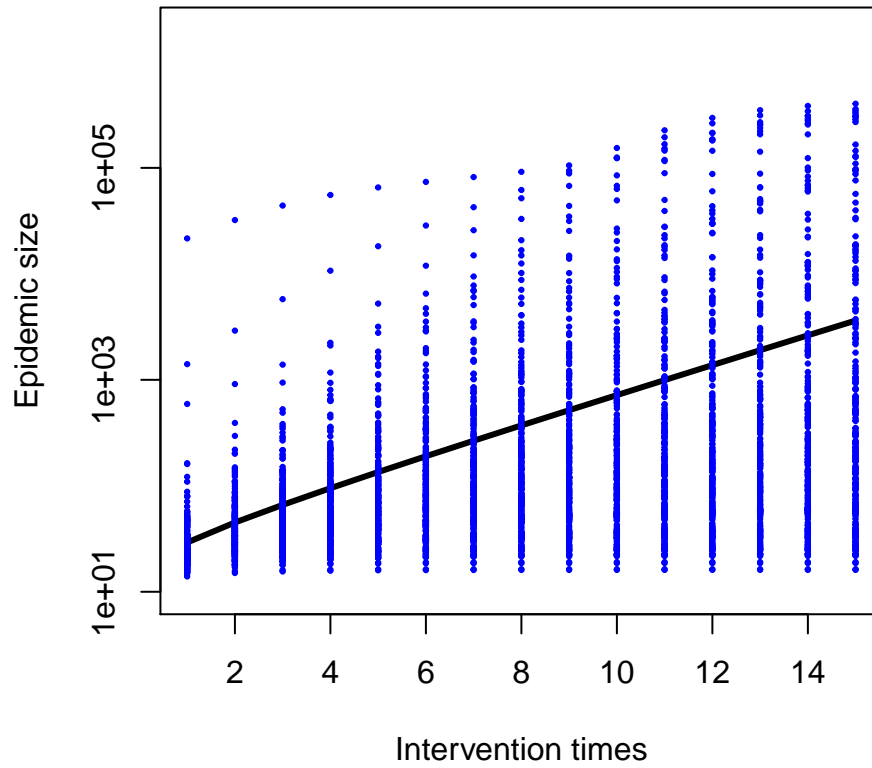
```


Finally, we set up a nested loop, first to cycle through different values of T , then to cycle through different simulated parameter sets. Note the pre-allocated data frame and use of the counter j , also to speed up evaluation.

```
> j <- 1
> data <- data.frame(matrix(rep(NA, levels*h2*14), nrow=levels*h2))
> for(i in 1:h2){
+   for (T in intervention.times){
+     data[j,1:13] <- params <- as.list(c(params.set[i,], T=T))
+     out <- as.data.frame(ode(xstart, times, legrand2, params))
+     data[j,14] <- get.size(out)
+     j <- j+1
+   }
+ }
> names(data) <- c(names(params), 'outbreak.size')
> save(data, file='data.Rdata')
```

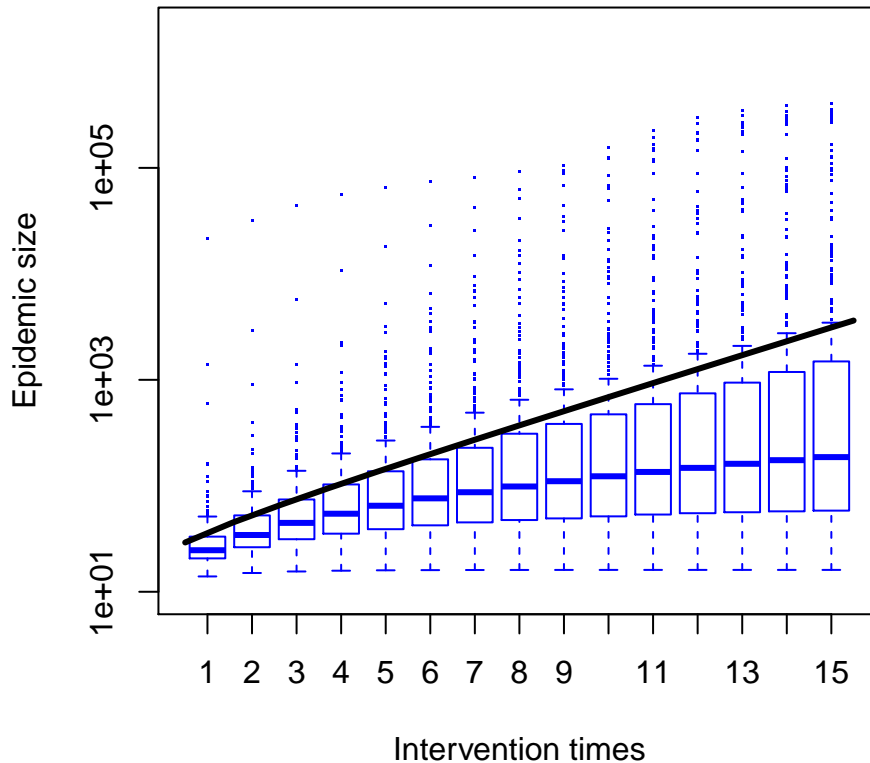
What do the results look like? We can plot each simulated value as a point and compare this with the estimate from Legrand et al.

```
> load('data.Rdata')
> plot(intervention.times, output0, type='l', lwd=3, ylim=c(10,2e6), log='y',
+       xlab='Intervention times',
+       ylab='Epidemic size')
> points(data$T, data$outbreak.size, pch=19, cex=0.3, col='blue')
```



This is somewhat difficult to interpret, however, so we will summarize these points with a box-and-whisker plot.

```
> boxplot(data$outbreak.size~data$T, ylim=c(10,2e6), border='blue', log='y', pch='.')
> par(new=TRUE)
> plot(intervention.times, output0, type='l', lwd=3, ylim=c(10,2e6), log='y', xlab='Intervention times')
```



Evidently, our best guesses at parameter values are somewhat pessimistic.

Exercise 4. In R, the function `cor` may be used to calculate the correlation between two sets of numbers. Write a script to determine which variable is most highly correlated with infections prevented.

Exercise 5. Having determined which variable total infections prevented is most sensitive to, re-visit the range of values that was considered for this parameter. Choose an alternative lower or upper bound and re-run the latin hypercube analysis. Does this change the result very much? What parameter would you recommend to be most important for further research?

One problem with the approach adopted here is that we're always changing more than one variable at a time. Is there any way to determine the effect of each parameter on the outcome while averaging over the other variables? Under such a thought experiment, can we identify which parameters are most important? If we may assume that the effect of a parameter on number of preventions is monotonic, then these questions can be answered by inspecting the partial rank correlations between each of the parameters and the number of infections prevented. Partial rank correlations can be computed using the `pcc` function in the R package `sensitivity`.

```
> library(sensitivity)
> bonferroni.alpha <- 0.05/12
> prcc <- pcc(data[,1:12], data[,14], nboot = 1000, rank=TRUE, conf=1-bonferroni.alpha)
> save(prcc, file='prcc.Rdata')
```

We can view a table of the resulting partial correlation coefficients. Since none of the (penalized) confidence intervals contains zero, we conclude that all are significant and produce a plot showing their relative magnitudes.

```
> library(sensitivity)
> load('prcc.Rdata')
> summary <- print(prcc)
```

Call:

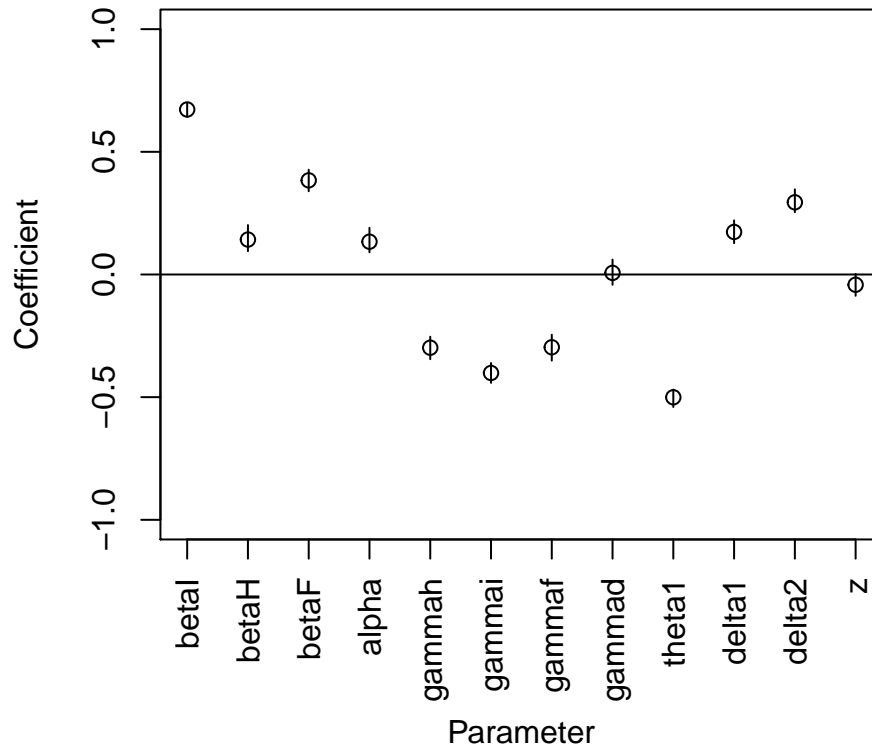
```
pcc(X = data[, 1:12], y = data[, 14], rank = TRUE, nboot = 1000, conf = 1 - bonferroni.alpha)
```

Partial Rank Correlation Coefficients (PRCC):

	original	bias	std. error	min. c.i.	max. c.i.
betaI	0.67248051	6.363399e-04	0.009761728	0.64310749	0.69791859
betaH	0.14237727	2.155177e-04	0.016735843	0.09516780	0.20156103
betaF	0.38375169	-1.800964e-04	0.015026960	0.33930083	0.42711776
alpha	0.13365564	-1.847920e-03	0.015960506	0.09067389	0.19036413
gammah	-0.29843298	-4.558288e-04	0.015630247	-0.34523367	-0.25363759
gammah	-0.40145192	8.213968e-05	0.014661182	-0.44181729	-0.36072851
gammaf	-0.29665329	8.884918e-05	0.015991029	-0.35103025	-0.24531861
gammad	0.00655676	-3.198202e-04	0.017253428	-0.04209687	0.06078493
theta1	-0.50054157	6.728586e-04	0.012576595	-0.54006219	-0.46892242
delta1	0.17332982	-5.273522e-04	0.015414134	0.12755639	0.22056686
delta2	0.29419951	-1.087162e-03	0.016020105	0.25375436	0.34707571
z	-0.04187132	7.526227e-04	0.016298982	-0.08674951	0.00238828

```
> par(mar=c(7,4,4,2)+0.1)
> plot(summary$original, main='Partial rank correlation coefficients', ylim=c(-1,1),
+       xlab='', ylab='Coefficient',
+       axes=FALSE)
> axis(2)
> axis(1, at=seq(1:12), labels=row.names(summary), las=2)
> mtext(text='Parameter', side=1, line=4.5)
> box()
> for(i in 1:12) lines(c(i,i),c(summary[i,4], summary[i,5]))
> abline(h=0)
```

Partial rank correlation coefficients



References

- Alexander, K.A., Sanderson, C.E., Marathe, M., Lewis, B.L., Rivers, C.M., Shaman, J., Drake, J.M., Lofgren, E., Dato, V.M., Eisenberg, M.C. & Eubank, S. (2015). What Factors Might Have Led to the Emergence of Ebola in West Africa? *PLoS neglected tropical diseases*, 9, e0003652.
- Legrand, J., Grais, R.F., Boelle, P.Y., Valleron, A.J. & Flahault, A. (2007). Understanding the dynamics of Ebola epidemics. *Epidemiology and infection*, 135, 610–21.
- Oyok, T., Odonga, C., Mulwani, E., Abur, J., Kaducu, F., Akech, M., Olango, J., Onek, P., Turyanika, J., Mutyaba, I., Luwaga, H., Bisoborwa, G., Kaguna, A., Omaswa, F., Zaramba, S., Okware, S., Opio, A., Amandua, J., Kamugisha, J., Mukoyo, E., Wanyana, J., Mugero, C., Lamunu, M., Ongwen, G., Dip, E., Mugaga, M. & Kiyonga, C. (2001). Outbreak of Ebola Hemorrhagic Fever —Uganda, August 2000–January 2001. *Morbidity and Mortality Weekly Report*, 50, 73–77.