<div align="center">

Mathematics 4MB3/6MB3 Mathematical Biology
2016 ASSIGNMENT 4

Group Name: The Infective Collective

Group Members: Aurora Basinski-Ferris, Michael Chong, Daniel Park, Daniel Presta

</div>

This assignment was due on Wednesday 14 March 2018 at 11:30am.

1 Time Series analysis of Recurrent Epidemics

(a) You should have received the following data files by e-mail:

<div align="center">

meas_uk__lon_1944-94_wk.csv
meas_uk__lpl_1944-94_wk.csv

</div>

These plain text comma-separated-value files list weekly cases of measles (in London and Liverpool, England, from 1944 to 1994). Depending on which research direction you select, you might receive other files in the same `ymdc` (year,month,day,count) format, where the count column might contain cases or deaths, for example. Write the following Ⓡ functions:

(i) `read.ymdc()`. Read a file in `ymdc` format and return a data frame containing these data and including a `date` column that has Ⓡ's `Date` class. The first (and potentially only) argument to this function should be the `filename` of the data file to be read.

Below, we create a function which returns a data frame with a date column and an associated data column. This is done by mutating the initial data frame from the csv file. We string together the year, month, and day that are originally in three separate columns, and then make this string part of the Date class.

```
library(tidyverse) #load necessary packages
library(stringr)
read.ymdc <- function(filename) {
  df<- read.csv(filename, skip=6) #skip 6 because 6 lines
  #at top of csv file that aren't data
  df %>% mutate(date= as.Date(str_c(year,month,day,sep='-'))) %>%
    select(-year,-month,-day)
}
```

(ii) `time.plot()`. Given a data frame produced using `read.ymdc()`, display the associated time plot. The first argument of the function should be the data frame. Further optional argument(s) should allow the user to smooth the time series with a moving average. By default, this function should create a new plot but there

should be an option to add to an existing plot. Implement this by having a logical `add` argument that is false by default (`add=FALSE`). This will allow you to add a smoothed version of the time series on top of the raw data, for example. The final argument should be the ellipsis (`...`) so that details such as colour and line style can be passed to the plotting commands used in this function.

Next, we create a function which produces a moving average time plot. We modify the data frame that the user supplies when calling the function such that each row is an average of the following s rows and the prior s rows (if s is the value that the user specifies to compute the moving average over). We also delete the first s rows and the last s rows. By default, the plot is separate from any previous plots that the user may have created, but you may add the plot to an existing plot by inputting 'TRUE' in the third argument of the function.

```
time.plot <- function(df, s, add=FALSE, ...) {
  df2 <- df
  for (i in (s+1):(nrow(df2)-s)) {
  df2$cases[i]<-mean(df$cases[(i-s):(i+s)])
  }
  df2$cases[1:s]<-NA
  df2$cases[(nrow(df2)-s):(nrow(df2))] <-NA

  if(add) {
    lines(df2$date,df2$cases, ...)
  } else {
  plot(df2$date, df2$cases,type="l", ...)
  }
}
```

(iii) `periodogram()`. Given a data frame produced using `read.ymdc()`, display the associated *period periodogram* (power spectrum as a function of period). The first argument of the function should be the data frame. By default, the entire time series should be used, but optional argument(s) should allow the user to specify a time range of interest. Use ℝ's `spectrum()` function to compute the power spectrum. Have `add` and `...` arguments as in `time.plot()`. Note that if `v` is a vector containing a time series of interest, you can obtain and plot its *frequency* periodogram as follows.

Finally, we create a function which produces a periodogram on a specific date range of a time series. By default, we have that the begin date is the beginning of the time series and the end date is the end of the time series. However, these can be modified by the user. The function filters the data frame to only include the date range specified by the user. Then it performs a periodogram on that filtered data frame using the commands given in the assignment. Similar to the time.plot function, by default, the plot is separate from any previous plots the user may have created, but the plot may be added to an existing plot by specifying 'TRUE' in the fourth argument of the function.

```
periodogram <-function(df,
                       start.date=as.Date(df$date[1]),
                       end.date=as.Date(df$date[nrow(df)]),
                       add=FALSE, ...) {
    df <- df %>%
     filter(date >= start.date & date<=end.date)

    s <- spectrum(df$cases, plot = FALSE)
    plot((s$freq)^(-1), s$spec, type='l', ...)
}
```

(b) Using your functions, make a multi-panel plot that clearly shows the temporal pattern of the time series and how its frequency structure changes over time. Think carefully about how to make this multi-panel figure as clear as possible for yourselves and your readers. Describe your figure, explaining what aspects of your figure you feel are puzzling or interesting and may be possible to understand using mechanistic mathematical modelling. (Repeat this for each of the epidemic time series you are given.)

First, we perform analysis on the London time series. Our first step is to plot the time series, as well as a plot smoothed by a 10 week moving average. This smoothed data helps us identify areas of the time series that appear to have similar patterns. Namely, we note that the beginning of the time series until around 1950 appears to have consistent patterns. We also note that the data from around 1950 to 1975 and 1975 until 1994 appear to have similar structure. To investigate these claims, we plot periodograms. Our first periodogram looks at the entire time series (we cut it off at 250 weeks as when we plotted the whole 2500 weeks, we stop seeing patterns after around 200 weeks). This first periodogram tells us that across the whole time series, the most power is at around the 100 week mark, and the next most power is located around the 50 week mark. We note that these roughly correlate to 2 and 1 year cycles. Next, to investigate how these cycles may change over the time series, we plot periodograms for the intervals mentioned earlier which we identified through the initial moving average plot. The plot in the upper right corner shows the periodogram for the period from the beginning of the time series until 1 January 1950. We see that in this time period, the most power is located around the 50 week mark. Next, we plot the time interval from 1 January 1950 until 1 January 1975. Here, we see that the most power is located around the 2 year mark, while there is also a significant amount of power located around the 1 year mark. This tells us that in this period, there are mostly 2 year cycles, with a less significant 1 year cycle. Finally, we look at the end of the time series (from 1 January 1975 until 31 December 1994). In this periodogram, we see that the most power is located around the 140 week mark, with the next most significant power around the 50 week mark. This suggests that in this time period, the most significant cycle is around a three year cycle. However, if we look at the periodogram for the whole time series, we see that that cycle is insignificant overall compared to the 1 year and 2 year cycles. This may be because the three year cycle only appares in the last part of the time series. It also may be because the magnitude of power behind this cycle is much lower than the other cycles in other sections of data.

Page 3 of 17

```
#load london time series using function from 1a(i)
londondata <- read.ymdc('meas_uk__lon_1944-94_wk.csv')
plot(londondata$date, londondata$cases, xlab="Weeks", ylab="Incidence", main ="10 Wee
#plot using 10 week moving average using function from 1a(ii)
time.plot(londondata,10,add= TRUE, col='red')
```

10 Week Moving Average Plot of Liverpool Data
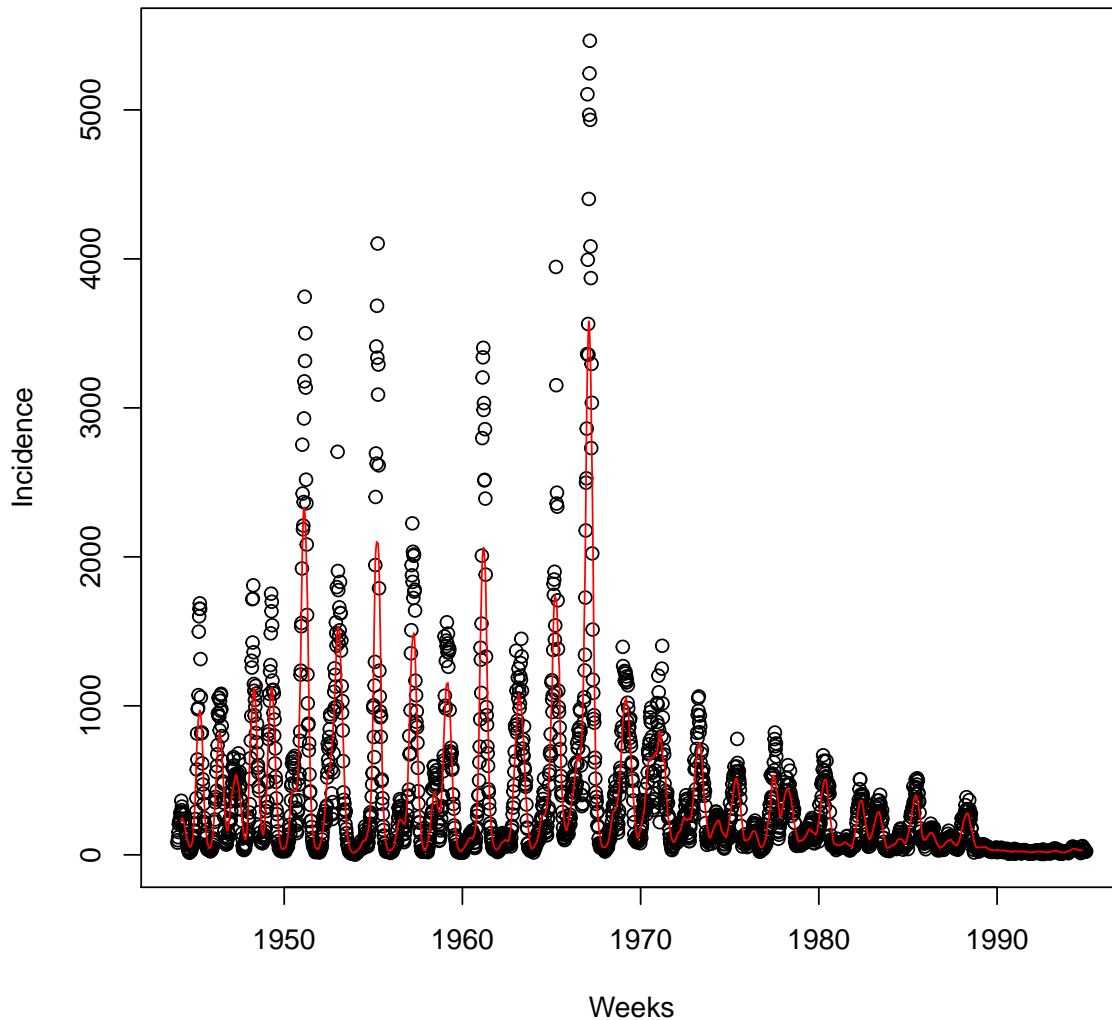

```
par(mfrow=c(2,2)) #create multiplot. two columns and two rows

periodogram(londondata,
            xlim=c(0,250),
            main="Entire time series",
            xlab="Weeks",
```

```r
            ylab="Power")

periodogram(londondata,
            end.date=as.Date('1950-01-01'),
            xlim=c(1,250),
            main="1944-01-07 to 1950-01-01",
            xlab="Weeks", ylab="Power")

periodogram(londondata,
            as.Date('1950-01-01'),
            as.Date('1975-01-01'),
            xlim=c(1,250),
            main="1950-01-01 to 1975-01-01",
            xlab="Weeks", ylab="Power")

periodogram(londondata,
            as.Date('1975-01-01'),
            as.Date('1994-12-31'),
            xlim=c(1,250),
            main="1975-01-01 to 1994-12-31",
            xlab="Weeks", ylab="Power")
```
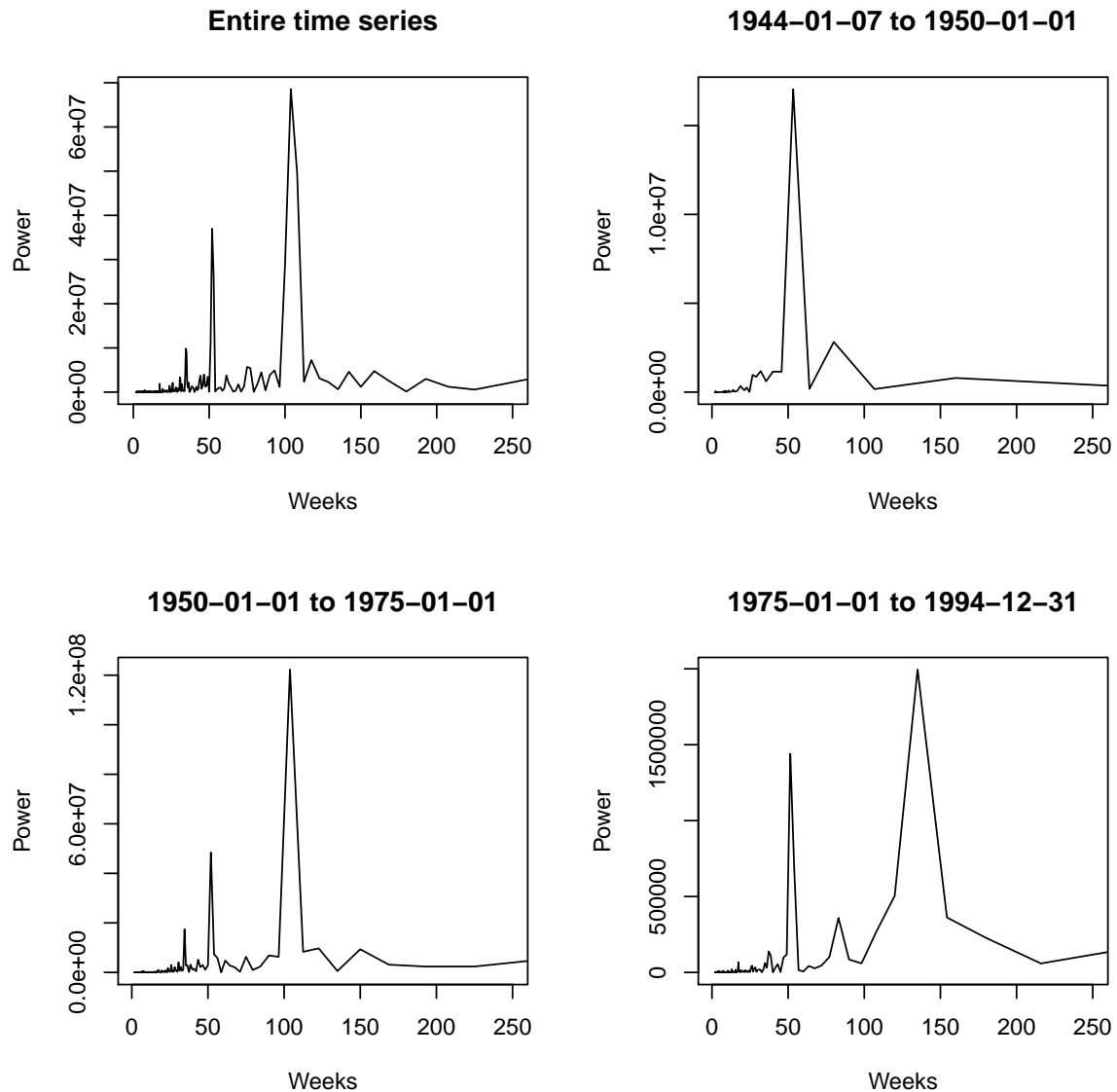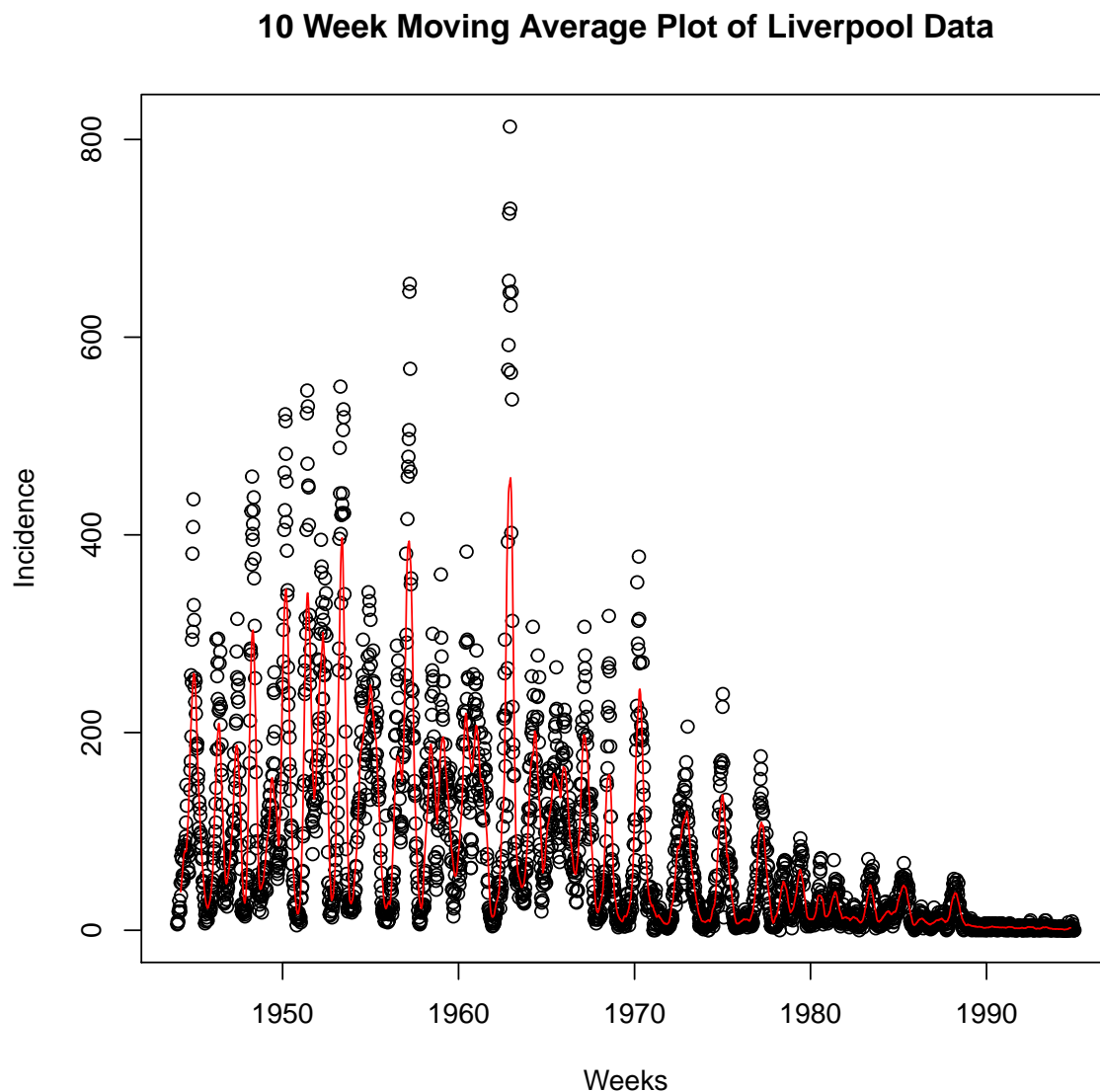
Entire time series

1944–01–07 to 1950–01–01

1950–01–01 to 1975–01–01

1975–01–01 to 1994–12–31

Next, we perform analysis on the Liverpool time series. Again, our first step is to plot the time series, as well as a plot smoothed by a 10 week moving average. Based on this smoothed data, we identify intervals of interest again, which will be tweaked and investigated with periodograms. These intervals identified are from the beginning of the data set to 1950, from 1950 until just before 1970, and from just before 1970 until the end of the time series. Our first plot is a periodogram of the whole time series. In this plot, we can see that the patterns across the whole time series aren't as clear as in the London data series. However, there appear to be strong 1 year and 2 year cycles despite some other less significant cycles present. We then investigate the intervals of interest which we identified through visual patterns in the moving average plot. First, we looked at the time interval from the beginning of the time series until 1 January 1950. The periodogram for this time revealed that the the strongest cycle in this time was around 1 year. Next, we found that the strongest cycle in the time period from 1 January 1950

until 1 January 1968 was two years. However, there was also a weak 1 year cycle present. Finally, in the time period from 1 January 1968 until 31 December 1994, we find that the most power is around the 130 week mark. However, there is also a significant amount of power around the 50 week mark.

```
#Perform analysis on the lpl time series
#load liverpool time series
lpldata<-read.ymdc('meas_uk__lpl_1944-94_wk.csv')
plot(lpldata$date, lpldata$cases, xlab="Weeks", ylab="Incidence", main="10 Week Movin
time.plot(lpldata,10,add=TRUE, col='red')
```

10 Week Moving Average Plot of Liverpool Data

```r
par(mfrow=c(2,2)) #create multiplot. two columns and two rows

#periodogram of full time series
periodogram(lpldata,
            main="Entire timeseries",
            xlim=c(0,300),xlab="Weeks", ylab="Power")
periodogram(londondata,
            end.date=as.Date('1950-01-01'),
            xlim=c(1,250),
            main="1944-01-07 to 1950-01-01",
            xlab="Weeks", ylab="Power")

#strongest 2 year cycle. slightly less strong 1 year cycle
periodogram(londondata,as.Date('1950-01-01'),
            as.Date('1968-01-01'),
            xlim=c(1,250),
            main="1950-01-01 to 1968-01-01",
            xlab="Weeks", ylab="Power")

#strong 3 year cycle. less strong 1 year cycle
periodogram(londondata,
            as.Date('1968-01-01'),
            as.Date('1994-12-31'),
            xlim=c(1,250),
            main="1968-01-01 to 1994-12-31",
            xlab="Weeks", ylab="Power")
```
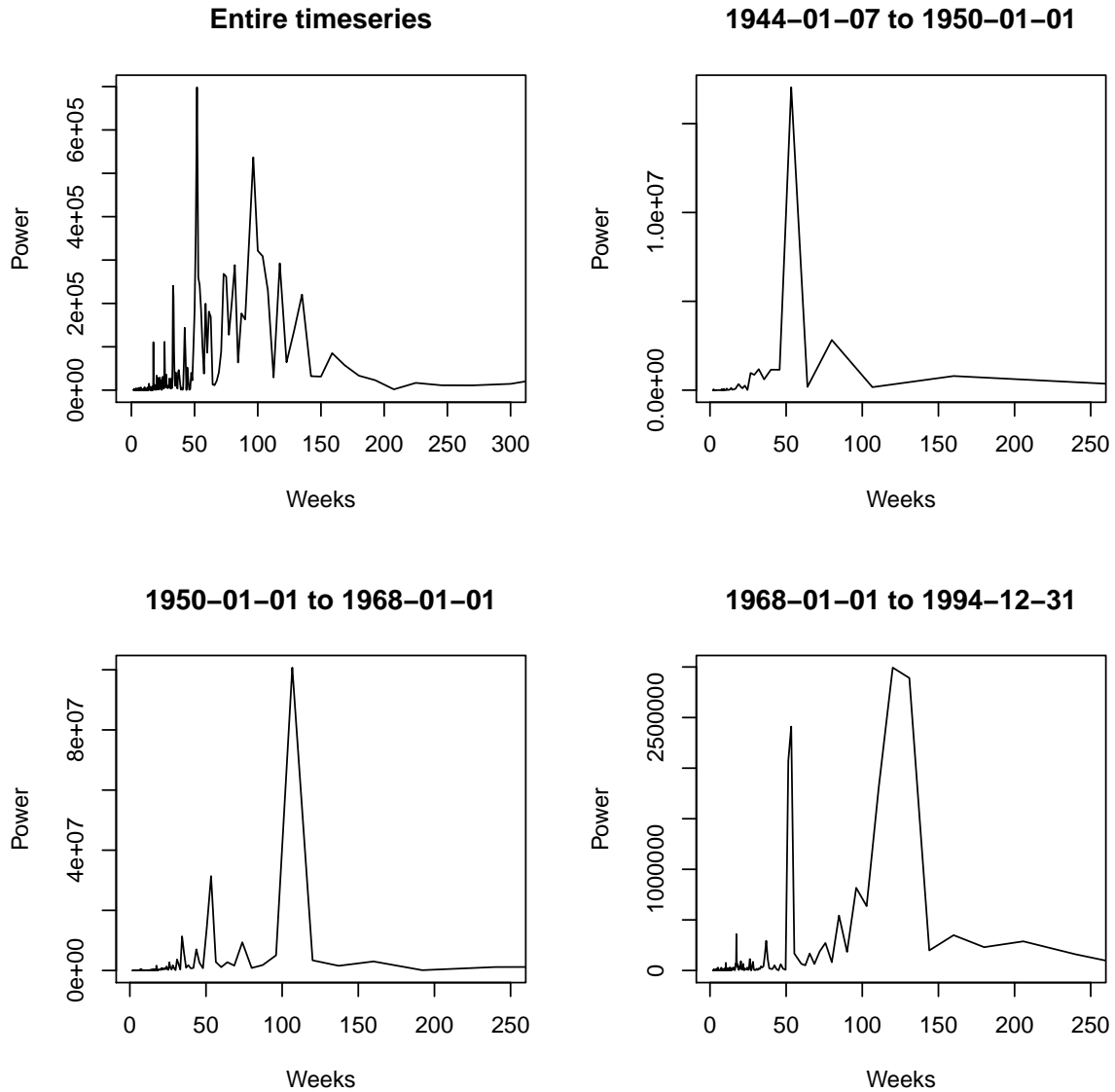
Entire timeseries

1944–01–07 to 1950–01–01

1950–01–01 to 1968–01–01

1968–01–01 to 1994–12–31

2 Stochastic Epidemic Simulations

Consider the SI model,

$$\frac{dI}{dt} = \beta(N - I)I\,, \qquad I(0) = I_0, \tag{1}$$

where β is the transmission rate, N is the population size and $I(t)$ is the number of infected individuals at time t.

(a) Write an Ⓡ function `SI.Gillespie()` that uses the Gillespie algorithm to produce a realization of a stochastic process whose mean field dynamics are given by equation (1) in the limit $N \to \infty$. Your function should have arguments `beta`, `N`, `I0` and `tmax` (the time at which to end the simulation). You may find it helpful (conceptually) to write

equation (1) in two-variable form:

$$\frac{dS}{dt} = -\beta SI, \qquad S(0) = N - I_0, \tag{2a}$$

$$\frac{dI}{dt} = \beta SI, \qquad I(0) = I_0. \tag{2b}$$

Note that there is only one type of event that can occur, so the second part of the Gillespie algorithm (what type of event occurred) is trivial for this model.

```r
SI.Gillespie <- function(beta = 1, N = 100, I0 = 1, tmax=10) {

  # initialize variables
  t <- 0
  I <- I0
  S <- N - I0
  n <- 1

  while(t[n] < tmax & I[n] < N) {
    # calculate event rate
    a <- beta*S[n]*I[n]

    # generate uniform random variable
    u <- runif(1)

    # time until next event
    dt <- (1/a)*log(1/(1-u))

    # record time
    t[n + 1] <- t[n] + dt

    # update state
    S[n+ 1] <- S[n] - 1
    I[n + 1] <- I[n] + 1

    # increment counter
    n <- n+1
  }

  # return time series
  tibble(
      time = t[1:(length(t) - 1)],
      infected = I[1:(length(I) - 1)],
      susceptible = S[1:(length(S) - 1)]
  )
```

```
}
```

(b) Make a multi-panel plot comparing the deterministic and stochastic dynamics of the SI model for $\beta = 1$, $I_0 = 1$ and $N \in \{32, 10^2, 10^3, 10^4\}$ ($N = 32$ is close to $10^{1.5}$). Each panel should correspond to a different value of N and should show 30 stochastic realizations together with the deterministic solution.

Note: To make stochastic simulations exactly reproducible use `set.seed()`.

```
library(deSolve)
# Function to describe
SI.d <- function(t, y, p) {
  with(as.list(c(y, p)), {
    dI <- beta*(N-I)*I

    return(list(c(dI)))
  })
}
```

```
set.seed(8)
test.sizes <- tribble(
  ~pop, ~tmax,
  32, 0.35,
  100, 0.15,
  1000, 0.02,
  10000, 0.0025
)

par(mfrow = c(2, 2))

initial <- c(I = 1)


for (i in 1:4) {
  size <- test.sizes$pop[i]
  tmax <- test.sizes$tmax[i]

  timesteps <- seq(0, tmax, length.out = 300)

  params <- c(beta = 1, N = size)
  sol <- ode(initial, timesteps, SI.d, params)
  plot(as.data.frame(sol), type = "l", lwd = 3, main= str_c("Solution of SI Model for
```
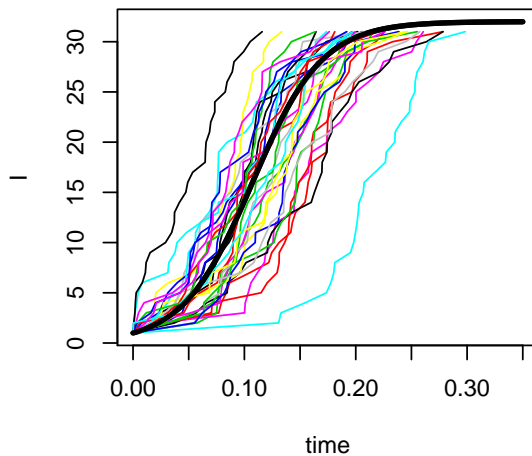
```
for (j in 1:30){
    lines(SI.Gillespie(beta = params["beta"],
                        N = params["N"],
                        I0 = initial["I"],
                        tmax = tmax), col = j)
}
lines(as.data.frame(sol), type = "l", lwd = 3)
}
```
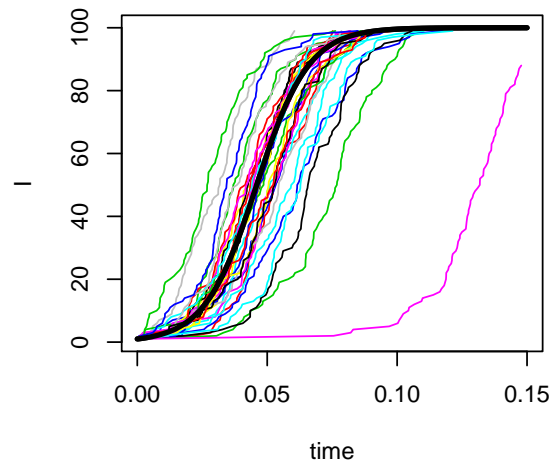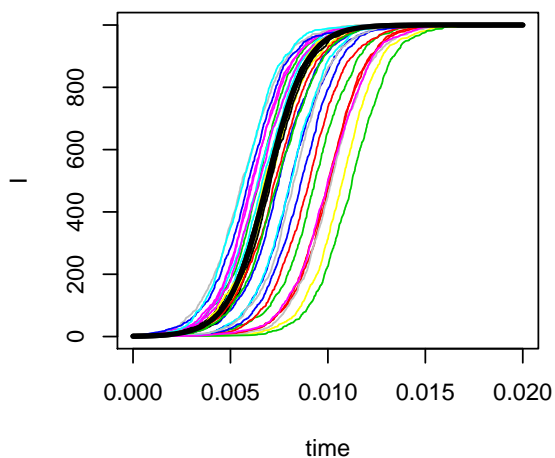
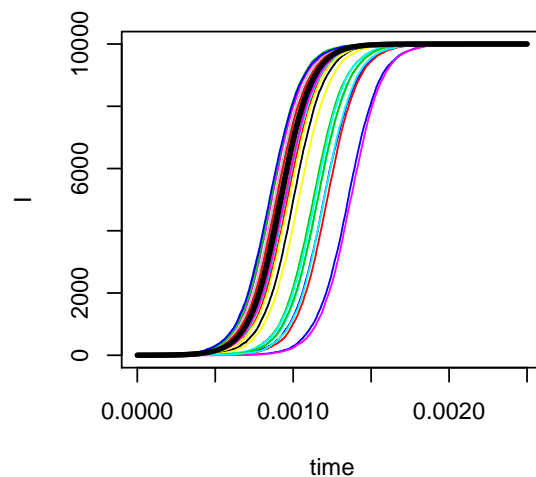**Solution of SI Model for N= 32**

Solution of SI Model for N= 100

Solution of SI Model for N= 1000

Solution of SI Model for N= 10000

The Gillespie simulation was run 30 times for each of the 4 population sizes: N = 32 (top left), N = 100 (top right), N = 1000 (bottom left) and N = 10000 (bottom right). Different runs of the simulation are shown in different colors, while the deterministic solution (using the `deSolve` package) is shown with a thick black line.

As N increases, we see that the Gillespie simulation becomes smoother, and agrees more closely with the deterministic solution in shape.

3 $\mathcal{R}_0$ for smallpox

The natural history of smallpox is shown in Figure 1. The US Centers for Disease Control and Prevention (CDC) has recently discovered that a group of bioterrorists plans to reintroduce smallpox to the United States. The CDC has reason to believe that the terrorists are also bioengineers and have successfully altered the virus so that it causes the early rash stage to be twice as long as it was when the virus was last circulating naturally in the 1970s. Moreover, the existing smallpox vaccine apparently provides no protection against the altered virus. The CDC wants your opinion on how the alterations to the virus will affect $\mathcal{R}_0$ and the expected final size of an epidemic if the planned attack is successful.

(a) Construct a compartmental (ODE) smallpox transmission model based on the natural history specified in Figure 1, including vital dynamics but ignoring disease-induced death.

Denote the susceptible state by S, the incubation state by E, and the recovered state by R. Furthermore, denote the four infectious stages by $I_1, I_2, I_3,$ and I_4, respectively, and let the each of the stages represent the four different levels of infectiousness exhibited by infected individuals. For example, the I_1 stage corresponds to the prodrom stage, in which infectiousness is rare, while I_4 stage represents both the pustules & scabs stage and the resolving scabs stage, as both of these stages share a low level of infectiousness. Note that all states are repsented by proportions.

Let σ represent the per capita rate at which an infected individual develops symptoms and let γ_i represent the per capita rate at which an infected individual in stage i progresses to the next stage. In other words, γ_i is considered to be the removal rate from stage I_i, such that the individual progresses to an infectiousness level exhibited in I_{i+1}. Moreover, note that γ_4 corresponds to the per capita recovery rate. Finally, let β_i represent the infectiousness (per contact transmission rate) of an infected individual in stage i and μ represent the per capita natural birth/death rate (they are assumed to be equal).

Then, we can write an ODE model for this system as follows:

$$\frac{dS}{dt} = \mu(1 - S) - (\beta_1 I_1 + \beta_2 I_2 + \beta_3 I_3 + \beta_4 I_4)S$$

$$\frac{dE}{dt} = (\beta_1 I_1 + \beta_2 I_2 + \beta_3 I_3 + \beta_4 I_4)S - (\sigma + \mu)E$$

$$\frac{dI_1}{dt} = \sigma E - (\gamma_1 + \mu)I_1$$

$$\frac{dI_2}{dt} = \gamma_1 I_1 - (\gamma_2 + \mu)I_1$$

$$\frac{dI_3}{dt} = \gamma_2 I_2 - (\gamma_3 + \mu)I_1$$

$$\frac{dI_4}{dt} = \gamma_3 I_3 - (\gamma_4 + \mu)I_1$$

$$\frac{dR}{dt} = \gamma_1 I_4 - \mu R$$

(b) Use a biological argument to find a formula for $\mathcal{R}_0$.

In order for an infected individual to infect a susceptible individual, it must survive the incubation period and become infectious. Then, we can think of $\mathcal{R}_0$ as the expected number of secondary cases caused by a typical infected individual, should they survive the incubation period and each of the following stages. This expression for $\mathcal{R}_0$ represents a sum of the basic reproductive numbers from each stage, such that the final number of secondary cases caused by the typical infective individual is merely equal to the number of infections they caused at each stage. For example, the contribution of infection from the first stage would be equivalent to $\mathcal{R}_0$ of an SEIR model. We denote this value as $\mathcal{R}_{0,1}$:

$$\beta_1 \times \frac{\sigma}{\sigma + \mu} \times \frac{1}{\gamma_1 + \mu}$$

where $\beta_1/(\gamma_1 + \mu)$ represent the average number of infections that occur in stage 1 and $\sigma/(\sigma+\mu)$ is the probability that an infected individual does not die before the incubation period is over. In order for infection to occur in stage $i > 1$, an infected individual must not die from natural mortality before reaching stage i. Then, the contribution of infection during the second stage would be defined as $\mathcal{R}_{0,2}$, such that

$$\beta_2 \times \frac{\sigma}{\sigma + \mu} \times \frac{\gamma_1}{\gamma_1 + \mu} \times \frac{1}{\gamma_2 + \mu}.$$

Note that we now have $\gamma_1/(\gamma_1 + \mu)$ to account for probability the of progressing to stage 2 without dying from natural causes while in stage 1. Likewise, we can do a similar computation for all other stages. Summing each of our values for $\mathcal{R}_{0,i}$, we ultimately

obtain

$$\mathcal{R}_0 = \beta_1 \times \frac{\sigma}{\sigma + \mu} \times \frac{1}{\gamma_1 + \mu}$$
$$+ \beta_2 \times \frac{\sigma}{\sigma + \mu} \times \frac{\gamma_1}{\gamma_1 + \mu} \times \frac{1}{\gamma_2 + \mu}$$
$$+ \beta_3 \times \frac{\sigma}{\sigma + \mu} \times \frac{\gamma_1}{\gamma_1 + \mu} \times \frac{\gamma_2}{\gamma_2 + \mu} \times \frac{1}{\gamma_3 + \mu}$$
$$+ \beta_4 \times \frac{\sigma}{\sigma + \mu} \times \frac{\gamma_1}{\gamma_1 + \mu} \times \frac{\gamma_2}{\gamma_2 + \mu} \times \frac{\gamma_3}{\gamma_3 + \mu} \times \frac{1}{\gamma_4 + \mu}.$$

(c) Calculate $\mathcal{R}_0$ using the next generation matrix approach. _Note:_ Your solution should include $\mathcal{F}$, $\mathcal{V}$, F, V, and FV^{-1}, in the most human-friendly form you can find. However, feel free to use symbolic manipulation software such as `Maple`, _Mathematica_ or sage to help with the necessary algebra and matrix computations. For this particular system, we have

$$\mathcal{F} = \begin{pmatrix} (\beta_1 I_1 + \beta_2 I_2 + \beta_3 I_3 + \beta_4 I_4)S \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \mathcal{V} = \begin{pmatrix} (\sigma + \mu)E \\ -\sigma E + (\gamma_1 + \mu)I_1 \\ -\gamma_1 I_1 + (\gamma_2 + \mu)I_1 \\ -\gamma_2 I_2 + (\gamma_3 + \mu)I_1 \\ -\gamma_3 I_3 + (\gamma_4 + \mu)I_1 \end{pmatrix}$$

Linearizing at the disease free equilibrium, we have

$$F = \begin{pmatrix} 0 & \beta_1 & \beta_2 & \beta_3 & \beta_4 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, V = \begin{pmatrix} (\sigma + \mu) & 0 & 0 & 0 & 0 \\ -\sigma & \gamma_1 + \mu & 0 & 0 & 0 \\ 0 & -\gamma_1 & \gamma_2 + \mu & 0 & 0 \\ 0 & 0 & -\gamma_2 & \gamma_3 + \mu & 0 \\ 0 & 0 & 0 & -\gamma_3 & \gamma_4 + \mu \end{pmatrix}$$

Inverting matrix V (by hand; steps not presnted), we observe:

$$V^{-1} = \begin{pmatrix} \frac{1}{\sigma+\mu} & 0 & 0 & 0 & 0 \\ \frac{\sigma}{(\sigma+\mu)(\gamma_1+\mu)} & \frac{1}{\gamma_1+\mu} & 0 & 0 & 0 \\ \frac{\sigma\gamma_1}{(\sigma+\mu)(\gamma_1+\mu)(\gamma_2+\mu)} & \frac{\gamma_1}{(\gamma_1+\mu)(\gamma_2+\mu)} & \frac{1}{\gamma_2+\mu} & 0 & 0 \\ \frac{\sigma\gamma_1\gamma_2}{(\sigma+\mu)(\gamma_1+\mu)(\gamma_2+\mu)(\gamma_3+\mu)} & \frac{\gamma_1\gamma_2}{(\gamma_1+\mu)(\gamma_2+\mu)(\gamma_3+\mu)} & \frac{\gamma_2}{(\gamma_2+\mu)(\gamma_3+\mu)} & \frac{1}{\gamma_3+\mu} & 0 \\ \frac{\sigma\gamma_1\gamma_2\gamma_3}{(\sigma+\mu)(\gamma_1+\mu)(\gamma_2+\mu)(\gamma_3+\mu)(\gamma_4+\mu)} & \frac{\gamma_1\gamma_2\gamma_3}{(\gamma_1+\mu)(\gamma_2+\mu)(\gamma_3+\mu)(\gamma_4+\mu)} & \frac{\gamma_2\gamma_3}{(\gamma_2+\mu)(\gamma_3+\mu)(\gamma_4+\mu)} & \frac{\gamma_3}{(\gamma_3+\mu)(\gamma_4+\mu)} & \frac{1}{\gamma_4+\mu} \end{pmatrix}$$

It is clear that matrix FV^{-1} consists of 0 entries except its first row. Hence, its eigenvalues are on its diagonal, four of which are zero. The only non-zero entry on the diagonal is the first column entry of the first row, which is equal to the previously derived $\mathcal{R}_0$

value:

$$\beta_1 \times \frac{\sigma}{\sigma + \mu} \times \frac{1}{\gamma_1 + \mu}$$

$$+ \beta_2 \times \frac{\sigma}{\sigma + \mu} \times \frac{\gamma_1}{\gamma_1 + \mu} \times \frac{1}{\gamma_2 + \mu}$$

$$+ \beta_3 \times \frac{\sigma}{\sigma + \mu} \times \frac{\gamma_1}{\gamma_1 + \mu} \times \frac{\gamma_2}{\gamma_2 + \mu} \times \frac{1}{\gamma_3 + \mu}$$

$$+ \beta_4 \times \frac{\sigma}{\sigma + \mu} \times \frac{\gamma_1}{\gamma_1 + \mu} \times \frac{\gamma_2}{\gamma_2 + \mu} \times \frac{\gamma_3}{\gamma_3 + \mu} \times \frac{1}{\gamma_4 + \mu}$$

Therefore, the derivation of $\mathcal{R}_0$ using the next generation method is consistent with the derivation of $\mathcal{R}_0$ from a biological argument.

(d) Based on your model, and $\mathcal{R}_0 \sim 5$ for unaltered smallpox, what can you say about the difference in $\mathcal{R}_0$ that can be expected for the newly engineered virus vs. the original virus?

Note that for unaltered small pox, the time scale of disease is much shorter than average life span of a person. Then, we can approximate $\mathcal{R}_0$ by assuming that $\mu \approx 0$. The expression for $\mathcal{R}_0$ thus becomes

$$\mathcal{R}_0 \approx \frac{\beta_1}{\gamma_1} + \frac{\beta_2}{\gamma_2} + \frac{\beta_3}{\gamma_3} + \frac{\beta_4}{\gamma_4}$$

The alteration causes the early rash stage to be twice as long and so γ_2^{-1} changes from 4 days to 8 days. It is evident that increasing γ_2^{-1} will lead to increase in $\mathcal{R}_0$.

Provided that infectiousness during the early rash stage is extreme, we can assume that at least half of the infection occurs during this stage. In the worst case scenario, all infections occur during the early rash stage. These assumptions are reasonable given that disease-induced death can occur in later stages, and so there would be little contribution to infection. Based on these assumptions, we have that

$$2.5 < \frac{\beta_2}{\gamma_{2,\text{original}}} < 5.$$

Since altering doubles γ_2^{-1}, we get

$$5 < \frac{\beta_2}{\gamma_{2,\text{altered}}} < 10.$$

This translates to

$$7.5 < \mathcal{R}_{0,\text{altered}} < 10.$$

(e) Write a paragraph that you can imagine e-mailing to the CDC, in which you do your best to answer their questions.

We highly recommend that the CDC begin preparing for the smallpox attack immediately. Given that smallpox has been altered to have twice as long of an early rash stage,

we expect $\mathcal{R}_0$ to increase by at least a factor of 1.5, and that is in the best-case scenario. In the worst case scenario, the $\mathcal{R}_0$ value corresponding to the altered smallpox will be twice as large as that of the unaltered smallpox, corresponding to the following range: $7.5 < \mathcal{R}_0 < 10$. This means that the final size of an epidemic will be almost 100% if the attack is successful.
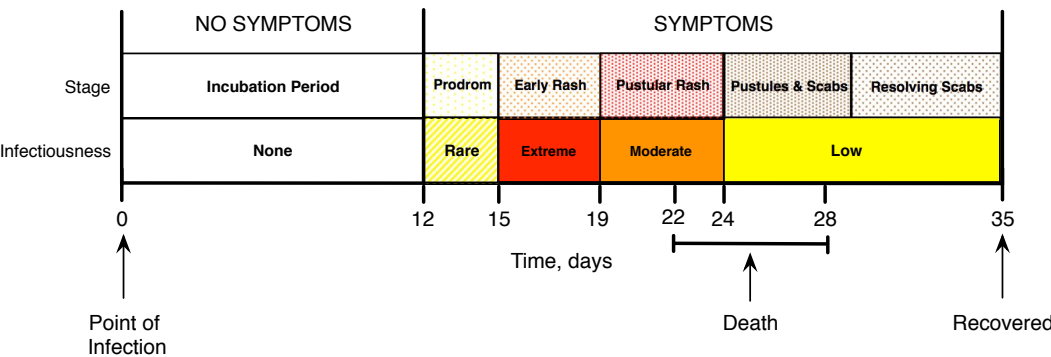
Figure 1: The natural history of smallpox infection. The prodrom stage begins with fever but the patient is very rarely contagious. Early rash is the most contagious stage, when the rash develops and transforms into bumps. During the pustular rash stage bumps become pustules, which then turn into scabs during the pustules and scabs stage and fall off during the resolving scabs stage. The infected person is contagious until the last scab falls off. (*This is Figure 3.4 from page 82 of Olga Krylova's 2011 McMaster University PhD thesis.*)

— END OF ASSIGNMENT —

Compile time for this document: March 13, 2018 @ 19:35