EEMB 179: Week 4: Homework

Kerri Luttrell

2/1/2021

PART THREE: HOMEWORK

Prep

```
library(deSolve)
```

Second, let's specify the initial conditions for our simulation

```
N <- 100   # population size; because dN/dt = 0, this doesn't change
I0 <- 1     # initial infected individual
S0 <- N - I0    # Assuming this is a new disease, everyone else is susceptible
R0 <- 0     # Assuming this is a new disease, there are no recovered individuals in the population
```

Third, we'll set up our storage variables. Note: Now we'll need one for each "state" of the population.

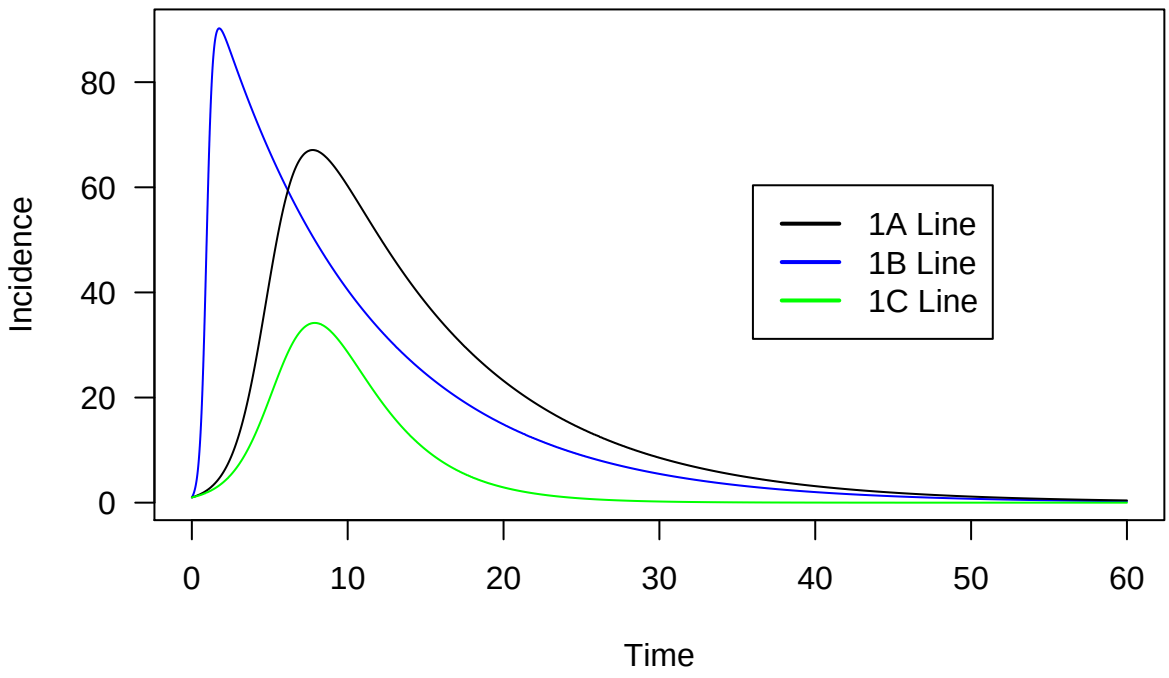
```
tset <- seq(from = 0, to = 60, length.out = 20000)# time steps
#set up our holding vectorss for each of our variables : N I S R
N.simu1 <- NaN*tset; N.simu1[1] <- N

I.simu1 <- NaN*tset#infectious
I.simu1[1] <- I0

S.simu1 <- NaN*tset#susceptible
S.simu1[1] <- S0

R.simu1 <- NaN*tset#recovered
R.simu1[1] <- 0

# semi colon separates two lines of code without phsyically separating them
N.simu.a <- NaN*tset; N.simu1[1] <- N
N.simu.b <- NaN*tset; N.simu1[1] <- N
N.simu.c <- NaN*tset; N.simu1[1] <- N

I.simu.a <- NaN*tset#infectious
I.simu.a[1] <- I0
I.simu.b <- NaN*tset#infectious
I.simu.b[1] <- I0
I.simu.c <- NaN*tset#infectious
I.simu.c[1] <- I0
```

```
S.simu.a <- NaN*tset#susceptible
S.simu.a[1] <- S0
S.simu.b <- NaN*tset#susceptible
S.simu.b[1] <- S0
S.simu.c <- NaN*tset#susceptible
S.simu.c[1] <- S0

R.simu.a <- NaN*tset#recovered
R.simu.a[1] <- 0
R.simu.b <- NaN*tset#recovered
R.simu.b[1] <- 0
R.simu.c <- NaN*tset#recovered
R.simu.c[1] <- 0
```

1. Make a single Incidence graph (plot number of infected individuals – I – over time) that overlays the following scenarios for a population of $N = 100$. Be sure to include a legend so that we can differentiate between them!

```
############################################################
# 1a. With a black line: beta = 0.01, gamma = 0.1, p = 0 #
############################################################
beta <- 0.01
gamma <- 0.1
p <- 0

for(i in 2:length(tset)){
    dt <- tset[i]-tset[i-1]
    S <- S.simu.a[i-1]
    I <- I.simu.a[i-1]
    R <- R.simu.a[i-1]
    dS <- (-beta*S*I)*dt
    dI <- (beta*S*I-gamma*I)*dt
    dR <- (gamma*I)*dt
    dN <- dS+dI+dR
    S.simu.a[i] <- S + dS
    I.simu.a[i] <- I + dI
    R.simu.a[i] <- R + dR
    N.simu.a[i] <- N + dN
}

############################################################
# 1b. With a blue line: beta = 0.05, gamma = 0.1, p = 0 #
############################################################
 beta.b <- 0.05

 for(i in 2:length(tset)){
    dt <- tset[i]-tset[i-1]
    S.b <- S.simu.b[i-1]
    I.b <- I.simu.b[i-1]
    R.b <- R.simu.b[i-1]
    dS.b <- (-beta.b*S.b*I.b)*dt
    dI.b <- (beta.b*S.b*I.b-gamma*I.b)*dt
    dR.b <- (gamma*I.b)*dt
```

```
    dN.b <- dS.b+dI.b+dR.b
    S.simu.b[i] <- S.b + dS.b
    I.simu.b[i] <- I.b + dI.b
    R.simu.b[i] <- R.b + dR.b
    N.simu.b[i] <- N + dN.b
 }

# ############################################################
# # 1c. With a green line: beta = 0.01, gamma = 0.3, p = 0 #
# ############################################################
 gamma.c <- 0.3


 for(i in 2:length(tset)){
    dt <- tset[i]-tset[i-1]
    S.c <- S.simu.c[i-1]
    I.c <- I.simu.c[i-1]
    R.c <- R.simu.c[i-1]
    dS.c <- (-beta*S.c*I.c)*dt
    dI.c <- (beta*S.c*I.c-gamma.c*I.c)*dt
    dR.c <- (gamma.c*I.c)*dt
    dN.c <- dS.c+dI.c+dR.c
    S.simu.c[i] <- S.c + dS.c
    I.simu.c[i] <- I.c + dI.c
    R.simu.c[i] <- R.c + dR.c
    N.simu.c[i] <- N + dN.c
 }

plot(x = tset, y = I.simu.b, type = 'l', las = 1, col = 'blue', xlab = 'Time', ylab='Incidence')
lines(x = tset, y = I.simu.a, col = 'black')
lines(x = tset, y = I.simu.c, col = 'green')
legend(x = max(tset)*0.6, y = max(I.simu.a)*0.9, legend = c('1A Line','1B Line','1C Line'), lwd = 2, col
```



/1 point black line

/1 point blue line

/1 point green line

/1 point legend

/1 point axes labels

= /5 points total

```
# 2. Describe the differences between the scenarios you plotted above.
#  2a. What is the effect of decreasing infectiousness?

# By decreasing infectivity of the pathogen and holding all else equal, the incidence of the disease is

#  2b. What is the effect of increasing recovery rates?
# By increasing recovery rates the incidence of the disease is decreased and the disease dies out of th
```

/2 points for the effect of decreasing infectiousness

/2 points for the effect of increasing infectiousness

= /4 points total

3. Calculate R_0 for each of each of the disease scenarios in Question 1, assuming an initial susceptible population size $S = 100$. What vaccination proportion (p) for each would be required to prevent a disease outbreak?

```r
#A
R_0.a <-  beta * N / gamma
p_crit.a <- 1 - 1/R_0.a
p_crit.a
```

```
## [1] 0.9
```

```r
#0.9

#B
R_0.b <-  beta.b * N / gamma
p_crit.b <- 1 - 1/R_0.b
p_crit.b
```

```
## [1] 0.98
```

```r
#0.98

#C
R_0.c <-  beta * N / gamma.c
p_crit.c <- 1 - 1/R_0.c
p_crit.c
```

```
## [1] 0.7
```

```r
#0.7

# Why are we using N in place of S in the above formulation?

# The disease is brand new so no members of the population have immunity to it at the initial timepoint
```

/2 points for work + answer for p_crit scenario 1

/2 points for work + answer for p_crit scenario 2

/2 points for work + answer for p_crit scenario 3

/1 points for $N \sim S$ answer

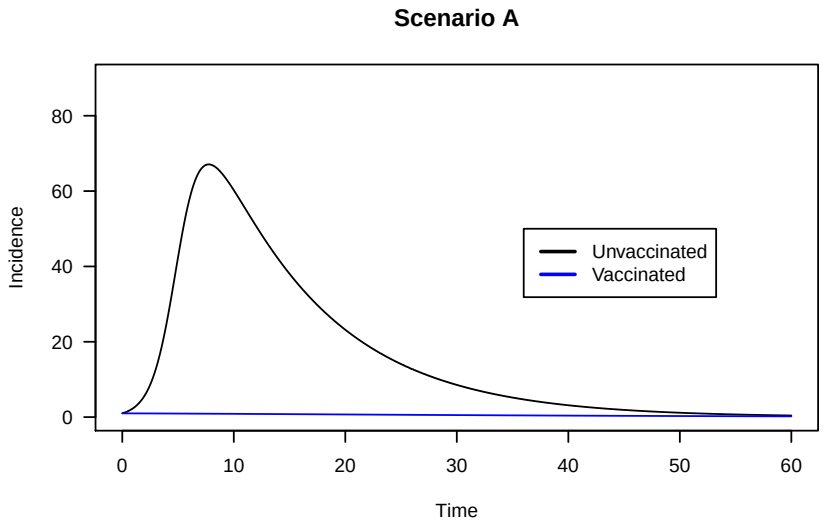
= /7 points total

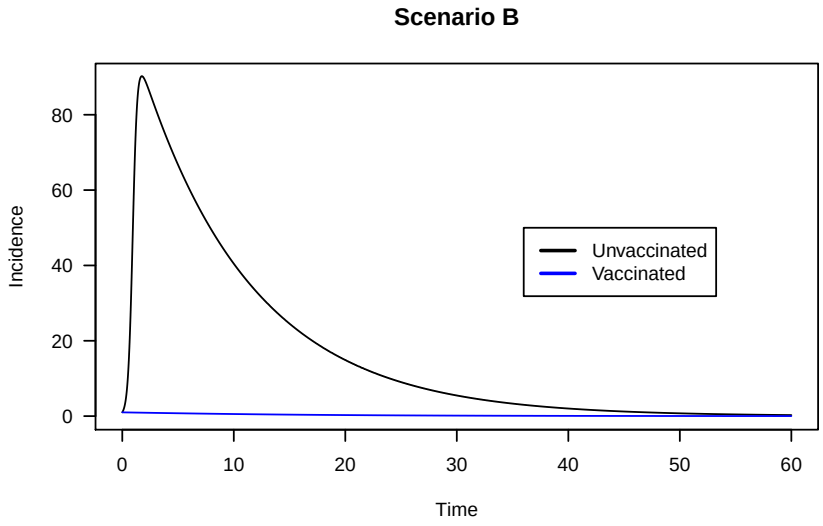
4. Run three simulations to check your answers for Part 3. For each, make an incidence diagram (total of 3 incidence diagrams) comparing the unvaccinated scenario (using a black line) with the vaccinated scenario (using a blue line). Don't forget to (1) label your plots so that we know which corresponds to which scenario, and (2) include a legend indicating which line corresponds to vaccinated vs. unvaccinated.

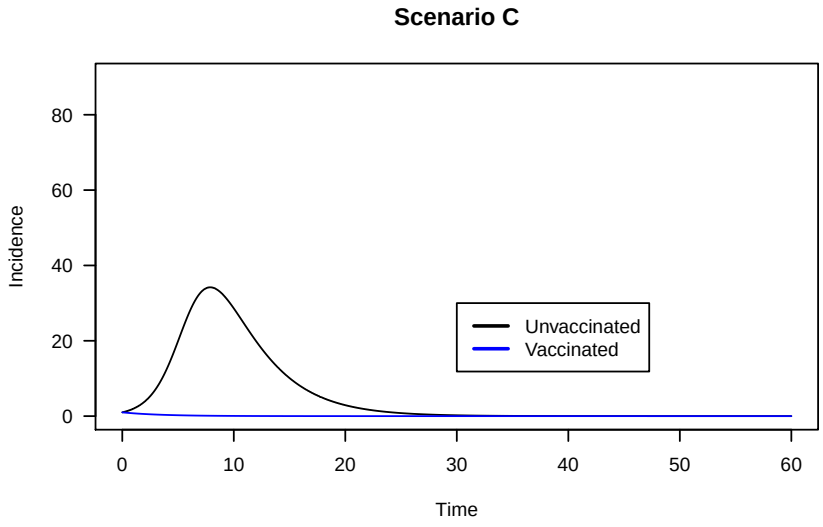
```r
par(mfcol = c(3, 1))
#SCENARIO A
VA <- N*p_crit.a
S0.v <- N - VA - I0
```

```
N.simu4.a <- NaN*tset; N.simu4.a[1] <- N
S.simu4.a <- NaN*tset; S.simu4.a[1] <- S0.v
I.simu4.a <- NaN*tset; I.simu4.a[1] <- I0
R.simu4.a <- NaN*tset; R.simu4.a[1] <- N - I0 - S0.v - VA

for(i in 2:length(tset)){
    dt <- tset[i]-tset[i-1]
    S.4a <- S.simu4.a[i-1]
    I.4a <- I.simu4.a[i-1]
    R.4a <- R.simu4.a[i-1]
    dS.4a <- (-beta*S.4a*I.4a)*dt
    dI.4a <- (beta*S.4a*I.4a-gamma*I.4a)*dt
    dR.4a <- (gamma*I.4a)*dt
    dN.4a <- dS.4a+dI.4a+dR.4a
    S.simu4.a[i] <- S.4a + dS.4a
    I.simu4.a[i] <- I.4a + dI.4a
    R.simu4.a[i] <- R.4a + dR.4a
    N.simu4.a[i] <- N + dN.4a
}

plot(x = tset, y = I.simu.a, type = 'l', las = 1, col = 'black', xlab = 'Time', ylab='Incidence', main=
lines(x = tset, y = I.simu4.a, col = 'blue')
legend(x = max(tset)*0.6, y = 50, legend = c('Unvaccinated','Vaccinated'), lwd = 2, col = c('black', 'bl

#SCENARIO B

VB <- N*p_crit.b
S0.vb <- N - VB - I0


N.simu4.b <- NaN*tset; N.simu4.b[1] <- N
S.simu4.b <- NaN*tset; S.simu4.b[1] <- S0.vb
I.simu4.b <- NaN*tset; I.simu4.b[1] <- I0
R.simu4.b <- NaN*tset; R.simu4.b[1] <- N - I0 - S0.vb - VA

for(i in 2:length(tset)){
    dt <- tset[i]-tset[i-1]
    S.4b <- S.simu4.b[i-1]
    I.4b <- I.simu4.b[i-1]
    R.4b <- R.simu4.b[i-1]
    dS.4b <- (-beta.b*S.4b*I.4b)*dt
    dI.4b <- (beta.b*S.4b*I.4b-gamma*I.4b)*dt
    dR.4b <- (gamma*I.4b)*dt
    dN.4b <- dS.4b+dI.4b+dR.4b
    S.simu4.b[i] <- S.4b + dS.4b
    I.simu4.b[i] <- I.4b + dI.4b
    R.simu4.b[i] <- R.4b + dR.4b
    N.simu4.b[i] <- N + dN.4b
}

plot(x = tset, y = I.simu.b, type = 'l', las = 1, col = 'black', xlab = 'Time', ylab='Incidence', main=
lines(x = tset, y = I.simu4.b, col = 'blue')
legend(x = max(tset)*0.6, y = 50, legend = c('Unvaccinated','Vaccinated'), lwd = 2, col = c('black', 'bl
```

```
?ylim
#SCENARIO C

VC <- N*p_crit.c
S0.vc <- N - VC - I0


N.simu4.c <- NaN*tset; N.simu4.c[1] <- N
S.simu4.c <- NaN*tset; S.simu4.c[1] <- S0.vb
I.simu4.c <- NaN*tset; I.simu4.c[1] <- I0
R.simu4.c <- NaN*tset; R.simu4.c[1] <- N - I0 - S0.vb - VA

for(i in 2:length(tset)){
    dt <- tset[i]-tset[i-1]
    S.4c <- S.simu4.c[i-1]
    I.4c <- I.simu4.c[i-1]
    R.4c <- R.simu4.c[i-1]
    dS.4c <- (-beta*S.4c*I.4c)*dt
    dI.4c <- (beta*S.4c*I.4c-gamma.c*I.4c)*dt
    dR.4c <- (gamma.c*I.4c)*dt
    dN.4c <- dS.4c+dI.4c+dR.4c
    S.simu4.c[i] <- S.4c + dS.4c
    I.simu4.c[i] <- I.4c + dI.4c
    R.simu4.c[i] <- R.4c + dR.4c
    N.simu4.c[i] <- N + dN.4c
}

plot(x = tset, y = I.simu.c, type = 'l', las = 1, col = 'black', xlab = 'Time', ylab='Incidence', main=
lines(x = tset, y = I.simu4.c, col = 'blue')
legend(x = max(tset)*0.5, y = 30, legend = c('Unvaccinated','Vaccinated'), lwd = 2, col = c('black', 'bl
```

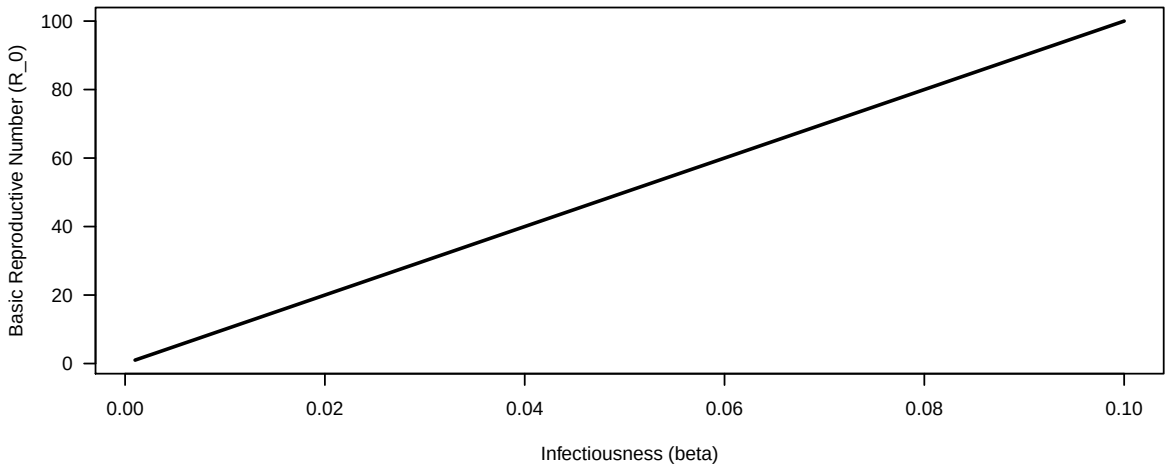

Scenario A
Incidence
Time
Unvaccinated
Vaccinated
0
20
40
60
80
0
10
20
30
40
50
60

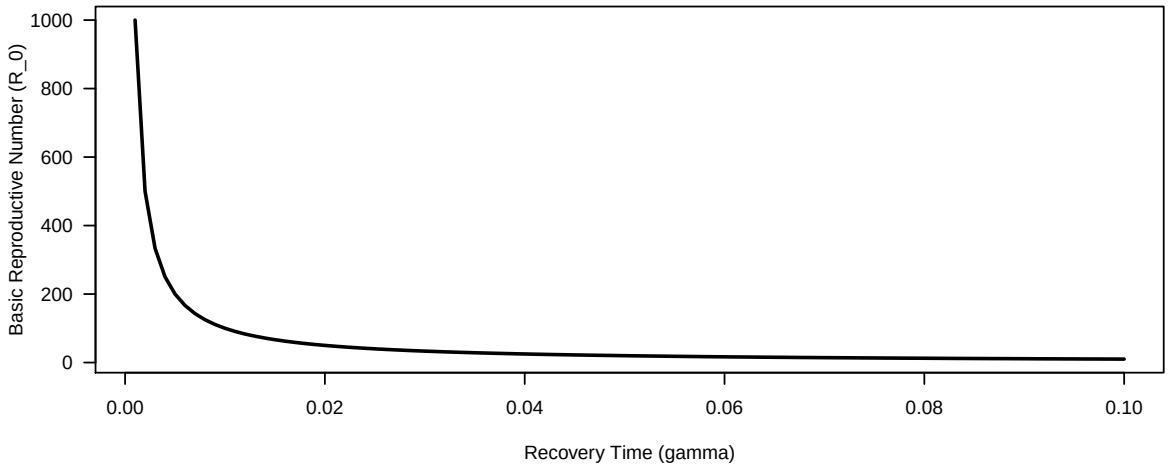

Scenario B
Incidence
Time
Unvaccinated
Vaccinated
0
20
40
60
80
0
10
20
30
40
50
60

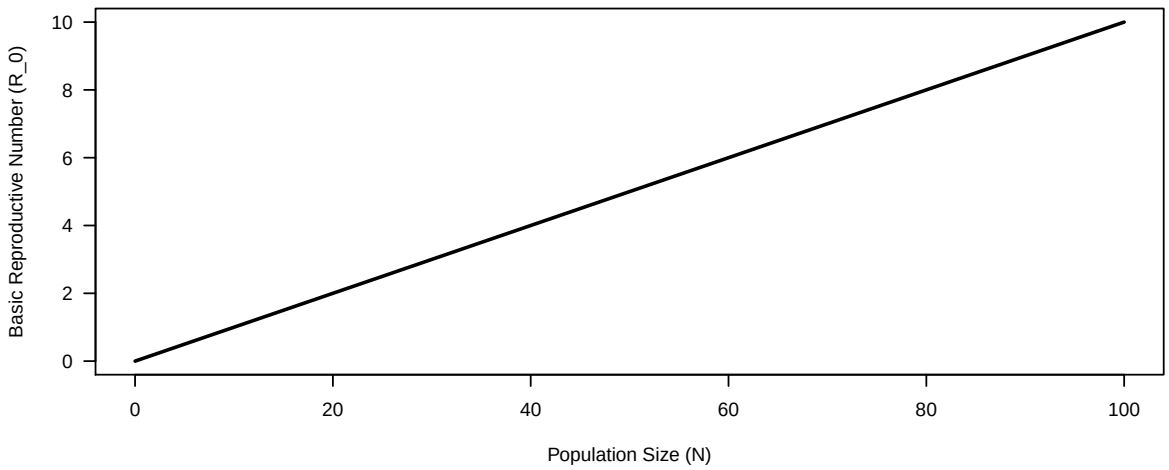

Scenario C
Incidence
Time
Unvaccinated
Vaccinated
0
20
40
60
80
0
10
20
30
40
50
60

```
/3 points for unvaccinated lines

/3 points for vaccinated lines

/3 points for legend

/3 points for axes

= /12 points total (3 plots)


 5. Explain how R_0 and p depend upon:
 R_0 =  beta * (N / gamma)
 p_crit = 1- 1/R_0

  5a. infectiousness,

When infectiousness increases, R0 increases asymptotically.
When infectiousness increases, p_crit increases asymptotically.

  5b. recovery time, and

When recovery time increases, R0 decreases linearly.
When recovery time increases, p_crit decreases linearly.

  5c. population size (assuming N = S, when the disease has not yet arrived).

When population size increseases, R0 increases asymptotically.
When population size increseases, p_crit increases asymptotically.

 For each of these (for a total of 6 plots -- don't forget to label the axes of each one!):
    - create a plot of R_0 (y-axis) vs. the parameter/variable of interest (x-axis),
  - create a plot of p (y-axis) vs. the parameter of interest (x-axis), and
    - explain the shapes of the graphs based on the biology of disease spread. Do you notice anything oc



'''r
beta_set <- seq(from = 0.001, to = 0.1, length.out = 100)
gamma_set <- seq(from = 0.001, to = 0.1, length.out = 100)
R0_set <- beta_set*N/gamma
Nset <- seq(from = 0, to = 100, length.out = 100)


par( mfrow= c(3,1))

#R_0 x Beta
plot(x = beta_set, y = R0_set, type = 'l', xlab = 'Infectiousness (beta)', ylab = 'Basic Reproductive Nu

R0g_set <- beta*N/gamma_set
# R_0 x Gamma
plot(x = gamma_set, y = R0g_set, type = 'l', xlab = 'Recovery Time (gamma)', ylab = 'Basic Reproductive
```

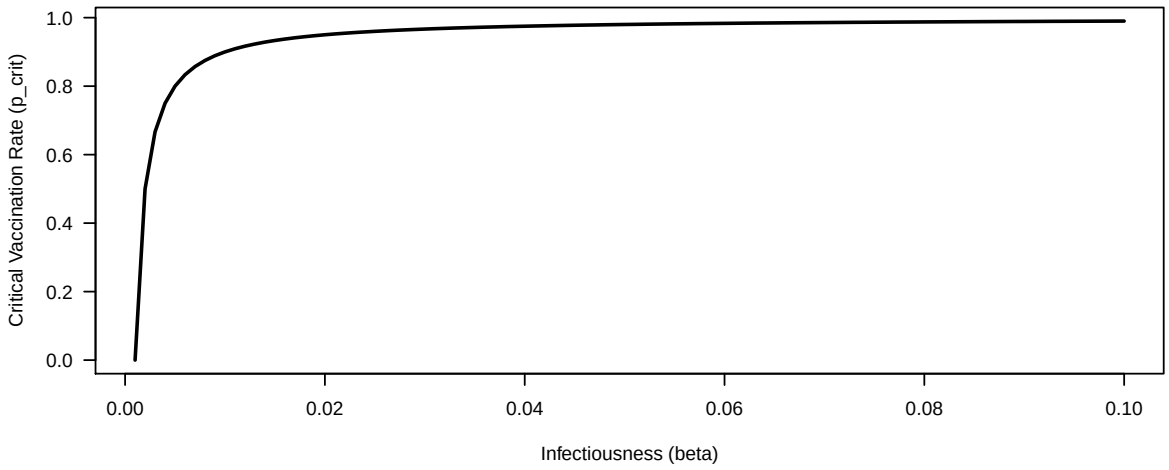
```
R0n_set <- beta*Nset/gamma
# R_0 x Population Size
plot(x = Nset, y = R0n_set, type = 'l', xlab = 'Population Size (N)', ylab = 'Basic Reproductive Number
```

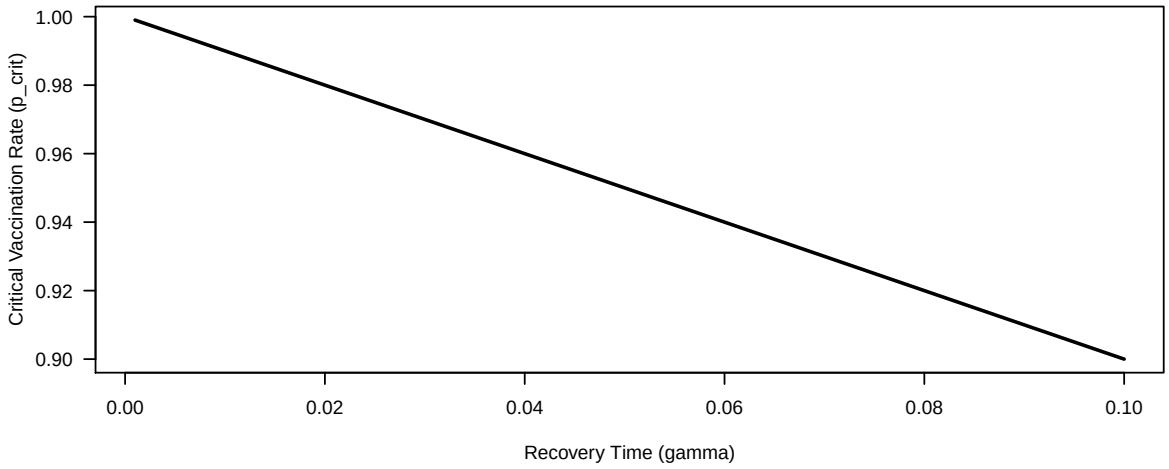

Basic Reproductive Number (R_0)
0
20
40
60
80
100
0.00
0.02
0.04
0.06
0.08
0.10
Infectiousness (beta)

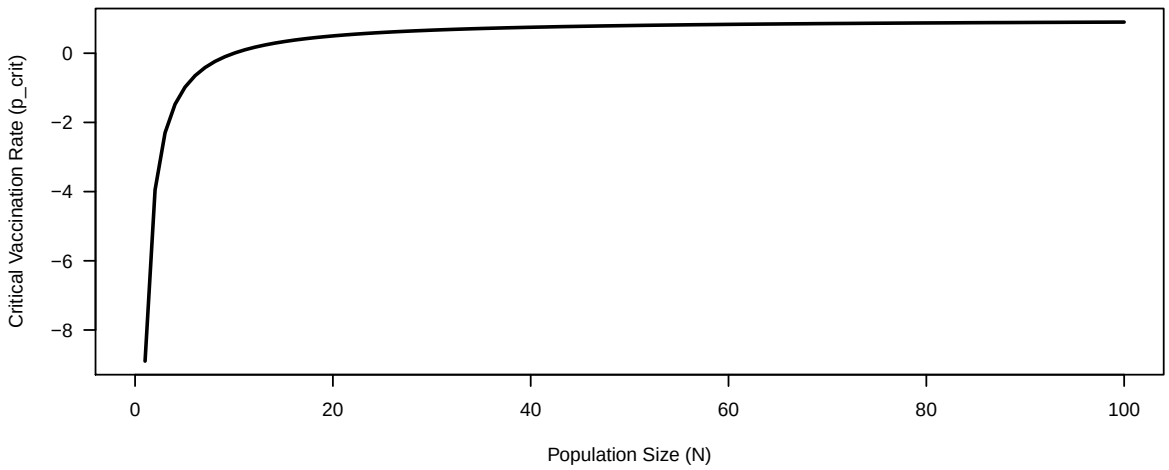

Basic Reproductive Number (R_0)
0
200
400
600
800
1000
0.00
0.02
0.04
0.06
0.08
0.10
Recovery Time (gamma)


Basic Reproductive Number (R_0)
0
2
4
6
8
10
0
20
40
60
80
100
Population Size (N)

```r
par( mfrow= c(3,1))
p_crit_set <- 1 - 1/R0_set

# p_crit x Beta
plot(x = beta_set, y = p_crit_set, type = 'l', xlab = 'Infectiousness (beta)', ylab = 'Critical Vaccina

p_critg_set <- 1 - 1/R0g_set

# p_crit x Gamma
plot(x = gamma_set, y = p_critg_set, type = 'l', xlab = 'Recovery Time (gamma)', ylab = 'Critical Vacci

p_critn_set <- 1 - 1/R0n_set
# p_crit x Pop Size
plot(x = Nset, y = p_critn_set, type = 'l', xlab = 'Population Size (N)', ylab = 'Critical Vaccination
```


Critical Vaccination Rate (p_crit)
1.0
0.8
0.6
0.4
0.2
0.0
0.00
0.02
0.04
0.06
0.08
0.10
Infectiousness (beta)


Critical Vaccination Rate (p_crit)
1.00
0.98
0.96
0.94
0.92
0.90
0.00
0.02
0.04
0.06
0.08
0.10
Recovery Time (gamma)


Critical Vaccination Rate (p_crit)
0
−2
−4
−6
−8
0
20
40
60
80
100
Population Size (N)

The pcrit values are negative for p_crit x Pop Size because at a low population size (N<10) with a beta of 0.01 and gamma of 0.1 the R0 value is less than 1. So, pcrit = 1 - 1/ RO would be a negative number.

/6 points for explanation of how R_0 and p depend upon infectiousness, recovery time, and population size.

/6 point per simulation line

/6 point per set of axes

/2 point for explanation of negative p_crit values in answer to 5c

= /20 points total (6 plots)