

Homework 2 (CMED 6210 Summer 2024)

PENG_XIN_3036274164_HW2

May 27, 2024

This homework is based on the paper “Transmission potential of smallpox in contemporary populations” by Gani et al (Nature 414: 748-751).

Consider a population size of 7 million. Assume that smallpox has $R_0 = 4.3$, mean latent period 14.5 days, and mean infectious period 8.5 days. Use the parameter values in Table 2 of Gani et al unless stated otherwise. Consider a time horizon of 365 days. Show your R script for each question.

Question 1: SEIR model with quarantine

Plot person-day quarantined and cumulative deaths as a function of the proportion of infections quarantined (as shown on our lecture slides). (30 points)

```
sim_SEIR_quarantine <- function(R0, psi, N, dur_latent, dur_infect, seed, tmax, dt, theta_values)
{
  # Function that takes model params and returns results
  # TODO: initialize and create variables
  alpha <- 1 / dur_latent
  gamma <- 1 / dur_infect
  beta <- R0 / (dur_infect * N * psi)
  times <- seq(0, tmax, by = dt)
  n_steps <- tmax / dt

  res <- data.frame()
  # TODO: simulate the model
  for (theta in theta_values) {
    S <- rep(NA, n_steps)
    E <- rep(NA, n_steps)
    I <- rep(NA, n_steps)
    U <- rep(NA, n_steps)
    Q <- rep(NA, n_steps)
    time <- rep(NA, n_steps)

    S[1] <- N - seed
    E[1] <- 0
    I[1] <- seed
    U[1] <- 0
    Q[1] <- 0
    time[1] <- 0

    for (i in 1:(n_steps)) {
```

```

dS_dt <- -beta * psi * I[i] * S[i]
dE_dt <- beta * psi * I[i] * S[i] - (1-theta)*alpha * E[i] - theta*alpha*E[i]
dI_dt <- (1-theta) * alpha * E[i] - gamma * I[i]
dU_dt <- gamma * I[i] + 0.04* Q[i]
dQ_dt <- theta * alpha * E[i] - 0.04 * Q[i]

S[i + 1] <- S[i] + dS_dt * dt
E[i + 1] <- E[i] + dE_dt * dt
I[i + 1] <- I[i] + dI_dt * dt
U[i + 1] <- U[i] + dU_dt * dt
Q[i + 1] <- Q[i] + dQ_dt * dt
time[i + 1] <- time[i] + dt
}

# TODO: collect results
res<-rbind(res, data.frame(time = time, S = S, E = E, I = I, U = U, Q = Q, theta = theta))

} # TODO: replace NA

res # this should be the last line in your function
}

```

Run simulations

```

# TODO: declare params
R0 <- 4.3
psi <- 0.5
N <- 7000000
dur_latent <- 14.5
dur_infect <- 8.5
seed <- 100
tmax <- 365
dt <- 1
theta_values <- seq(0, 1, by = 0.1)

# TODO: create variables to store results for plotting
res <- sim_SEIR_quarantine(R0, psi, N, dur_latent, dur_infect, seed, tmax, dt, theta_values)

# TODO: run model and record results
results_summary <- res %>%
  group_by(theta) %>%
  summarise(person_days_quarantined = sum(Q),
            cumulative_deaths = max(U))

```

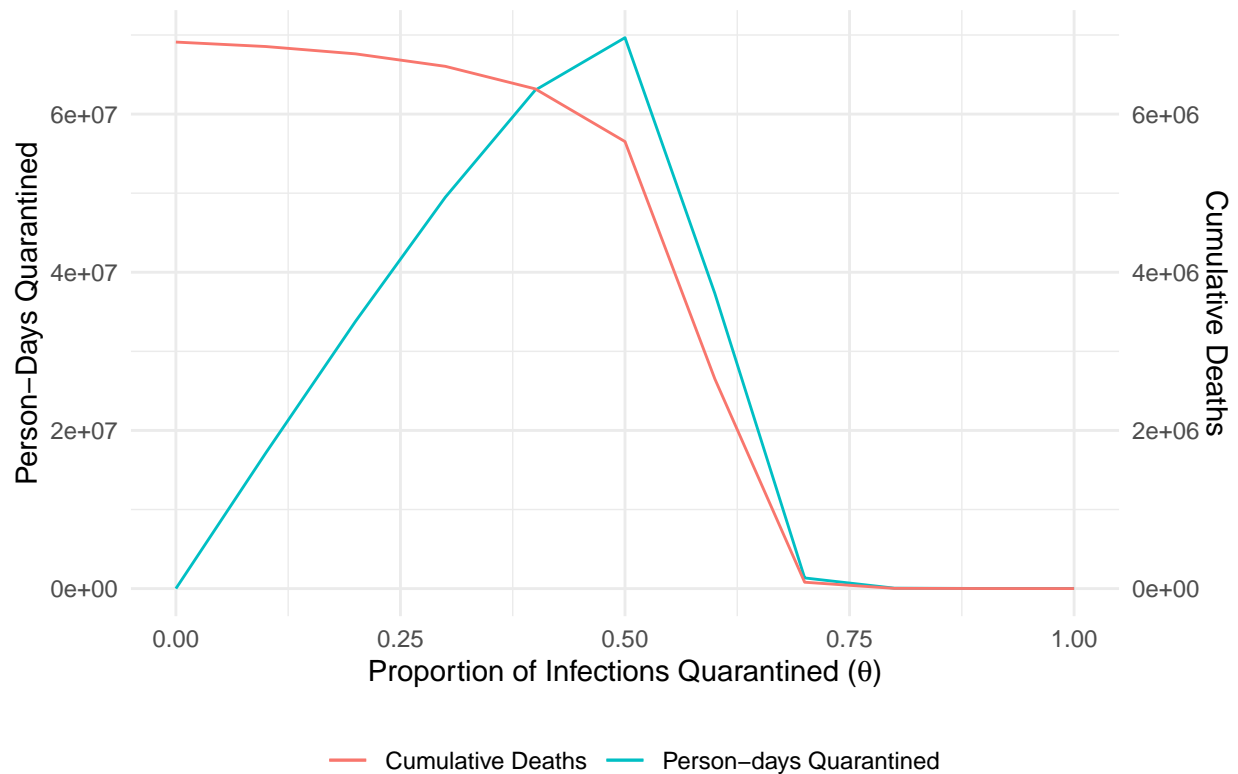
Plot person-day quarantined and cumulative deaths

```

# TODO: plot results
ggplot(results_summary, aes(x = theta)) +
  geom_line(aes(y = person_days_quarantined, color = "Person-days Quarantined")) +

```

```
geom_line(aes(y = cumulative_deaths * 1e1, color = "Cumulative Deaths")) +
scale_y_continuous(
  name = "Person-Days Quarantined",
  sec.axis = sec_axis(~ . / 1e1, name = "Cumulative Deaths")
) +
labs(title = "",
  x = expression("Proportion of Infections Quarantined (" * theta * ")"),
  color = "") +
theme_minimal() + theme(legend.position = 'bottom')
```



Question 2: SEIR with quarantine and contact tracing

Plot person-day disrupted and cumulative deaths as a function of the proportion of contacts traced assuming $\theta = 0.4$ (as shown on our lecture slides). (70 points)

```
sim_SEIR_quarantine_isolation <- function(R0, psi, N, dur_latent, dur_infect, seed, tmax, dt,
  rho_values) {
  # Function that takes model params and returns results
  # TODO: initialize and create variables
  alpha <- 1 / dur_latent
  gamma <- 1 / dur_infect
  beta <- R0 / (dur_infect * N * psi)
  times <- seq(0, tmax, by = dt)
  n_steps <- tmax / dt
```

```

res <- data.frame()

# TODO: simulate the model
for (rho in rho_values) {
  S <- rep(NA, n_steps)
  En <- rep(NA, n_steps)
  Ei <- rep(NA, n_steps)
  I <- rep(NA, n_steps)
  U <- rep(NA, n_steps)
  Q <- rep(NA, n_steps)
  C <- rep(NA, n_steps)
  V <- rep(NA, n_steps)
  time <- rep(NA, n_steps)

  S[1] <- N - seed
  En[1] <- 0
  Ei[1] <- 0
  I[1] <- seed
  U[1] <- 0
  Q[1] <- 0
  C[1] <- 0
  V[1] <- 0
  time[1] <- 0

  for (i in 1:(n_steps)) {
    dS_dt <- 0.06*(1-0.975)*C[i] - (1-rho)*beta * psi * I[i] * S[i] -
      rho*beta * psi * I[i] * S[i] - rho*beta*(1-psi)*I[i]*S[i]
    dEn_dt <- (1-rho) * beta * psi * I[i] * S[i] - (1-0.4)*alpha * En[i] - 0.4*alpha*En[i]
    dEi_dt <- rho*beta * psi * I[i] * S[i] - alpha*(1-0.3)*Ei[i] - 0.06*0.3*Ei[i]
    dI_dt <- (1-0.4) * alpha * En[i] - gamma * I[i]
    dU_dt <- gamma * I[i] + 0.04* Q[i]
    dQ_dt <- alpha*(1-0.3)*Ei[i] + 0.4 * alpha * En[i] - 0.04 * Q[i]
    dC_dt <- rho*beta*(1-psi)*I[i]*S[i] - 0.06*(1-0.975)*C[i]-0.06*0.975*C[i]
    dV_dt <- 0.06*0.3*Ei[i] + 0.06*0.975*C[i]

    S[i + 1] <- S[i] + dS_dt * dt
    En[i + 1] <- En[i] + dEn_dt * dt
    Ei[i + 1] <- Ei[i] + dEi_dt * dt
    I[i + 1] <- I[i] + dI_dt * dt
    U[i + 1] <- U[i] + dU_dt * dt
    Q[i + 1] <- Q[i] + dQ_dt * dt
    C[i + 1] <- C[i] + dC_dt * dt
    V[i+1]<-V[i]+dV_dt * dt
    time[i + 1] <- time[i] + dt
  }
  # TODO: collect results

  res <- rbind(res, data.frame(time = time, S = S, I = I, En=En, Ei=Ei, U = U, Q = Q, C = C, V=V,
                                rho = rho))
} # TODO: replace NA

res # this should be the last line in your function
}

```

Run simulations

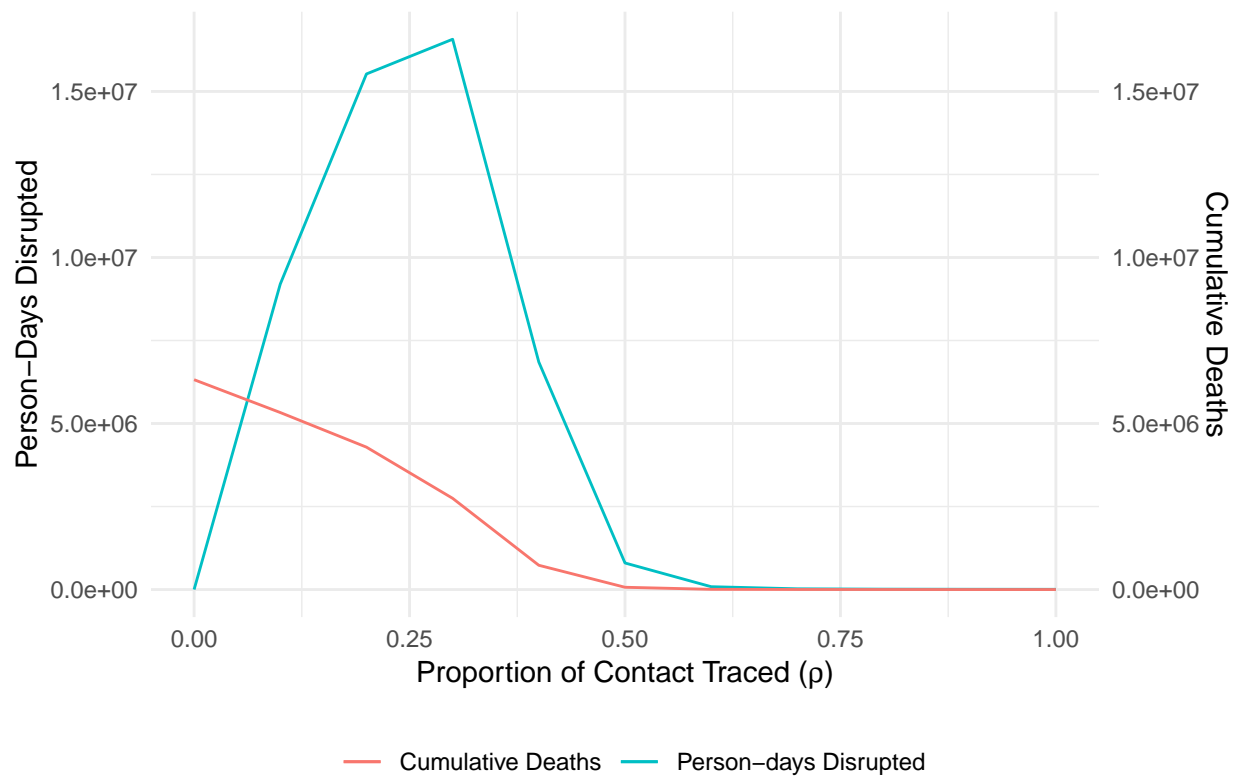
```
# TODO: declare params
R0 <- 4.3
psi <- 0.5
N <- 7000000
dur_latent <- 14.5
dur_infect <- 8.5
seed <- 100
tmax <- 365
dt <- 1
rho_values <- seq(0, 1, by = 0.1)

# TODO: create variables to store results for plotting
res <- sim_SEIR_quarantine_isolation(R0, psi, N, dur_latent, dur_infect, seed, tmax, dt, rho_values)

# TODO: run model and record results
results_summary <- res %>%
  group_by(rho) %>%
  summarise(person_days_disrupted = sum(C * dt),
            cumulative_deaths = max(U))
```

Plot person-day disrupted and cumulative deaths

```
# TODO: plot results
ggplot(results_summary, aes(x = rho)) +
  geom_line(aes(y = person_days_disrupted, color = "Person-days Disrupted", group=1)) +
  geom_line(aes(y = cumulative_deaths, color = "Cumulative Deaths", group=1)) +
  scale_y_continuous(
    name = "Person-Days Disrupted",
    sec.axis = sec_axis(~ ., name = "Cumulative Deaths")
  ) +
  labs(title = "",
       x = expression("Proportion of Contact Traced (*rho*)"),
       color = "") +
  theme_minimal() + theme(legend.position = 'bottom')
```



GenAI Statement AI Technologies were not used

I, Peng, Xin, 3036274164, hereby confirm that this assignment is the result of my own independent scholarly work, and that all material from the work of others have been fully acknowledged. No content generated by AI technologies has been used in this assessment