Intro to Infetious Diseases

Jesús N. Pinto-Ledezma and Jeannine Cavender-Bares

Short intro

Here add an INTRO

Set up your data and your working directory

```r
setwd("path/for/your/directory")
```

Install and load the following packages

```r
packages <- c("coronavirus", "deSolve", "dplyr", "tidyr", "ggplot2", "lubridate")
```

```r
# Install packages not yet installed
installed_packages <- packages %in% rownames(installed.packages())

if (any(installed_packages == FALSE)) {
  install.packages(packages[!installed_packages], dependencies = TRUE)
}
```

Loading all packages at once using **lapply** function.

```r
lapply(packages, library, character.only = TRUE)
```

Data exploration

First we will explore the data for the **COVID-19** for different countries. To do that we will get data at global scale using the amazing R package {**coronavirus**} that provides a daily summary of **COVID-19** cases by country obtained from the Johns Hopkins University Center for Systems Science and Engineering (JHU CCSE) Coronavirus.

```r
data("coronavirus")
```

To to get the most updated dataset we can use the function **refresh_coronavirus_jhu()** and store the UPDATED data in our **Environment** as a new object named *corona*.

```r
corona <- refresh_coronavirus_jhu()
```

```r
head(corona)
```

Let's start exploring the total numbers of confirmed cases by country.

```r
# Get top confirmed cases by country
corona_total <- coronavirus %>%
  filter(type == "confirmed") %>%
  group_by(country) %>%
  summarise(total = sum(cases)) %>%
  arrange(-total)

# See the 20 countries with more cases
head(corona_total, 20)
```

Now let's plot the top 20 countries with more reported cases of COVID-19

```r
corona_total %>%
  head(20) %>%
  ggplot(aes(y = country, x = total)) +
  geom_bar(stat = "identity") +
  labs(x = "Number of cases", y = "Countries with more reported cases")
```

What about the countries with less number of reported cases?

```r
corona_total %>%
  tail(20) %>%
  ggplot(aes(y = country, x = total)) +
  geom_bar(stat = "identity") +
  labs(x = "Number of cases", y = "Countries with less reported cases")
```

What about the number of reported recovered cases?

```r
# Get the number of recovered cases
coronavirus %>%
  filter(type == "recovered") %>%
  group_by(country) %>%
  summarise(total = sum(cases)) %>%
  arrange(-total) %>%
  head(20) %>%
  ggplot(aes(y = country, x = total)) + # plot the top 20 countries with more recovered subjects
  geom_bar(stat = "identity") +
  labs(x = "Number of recovered subjects", y = "Countries with more recovered subjects")
```

By quickly exploring the coronavirus dataset we can see that the United States is the country with more reported cases, not new information is added to our current knowledge, however, we can learn more about the COVID-19 dynamics in the US by looking at its changes over time–you can select any other country if you wish.

```r
corona_us <- subset(corona, location == "US")
head(corona_us)

corona_us <- corona_us[order(corona_us$date), ] # sort the data according dates
head(corona_us)
```

If we look at the US coronavirus data it contains information of the number of recovers (**R**), the number of new cases (**I**) and the the deaths. Let's isolate those data and inspect if there are some trends.

```r
infected_us <- subset(corona_us, data_type == "cases_new")
deaths_us <- subset(corona_us, data_type == "deaths_new")
recovered_us <- subset(corona_us, data_type == "recovered_new")

head(infected_us)
```

```r
plot(infected_us$date, infected_us$value, type = "b")
```

Ugh, ugly figure, let's try adding a new column that represent the data increasing since the first day in this dataset (i.e., 22/01/2020).

```r
Days <- 1:nrow(infected_us)
infected_us <- data.frame(infected_us, Days)
head(infected_us)
```

```r
plot(infected_us$Days, infected_us$value, type = "b",
     ylab = "Infected", xlab = "Days since the first case")
```

We can see that there is trend in the number of confirmed Infected subjects since the first case registered in the US, happily, the number of reported cases are declining steadily.

How do you feel about that? Please explain the trend in a temporal way, in other words, the pikes in the number of cases match with the US holidays? What about the number of recovered subjects?

SIR model

Data preparation

The first step to build our first ***SIR*** model is to get data of a disease, well, we already have data of coronavirus for each country in the world, however, as this dataset is at country level we probably will be missing some relevant information that occurs only at local scale, i.e., by States. In this sense, we will use data from the **Centers for Disease Control and Prevention** that reports aggregate counts of COVID-19 at State level within the United States, how good is that? Okay, let's go for it!

```r
# Get data from the CDC
url_data <- "https://data.cdc.gov/api/views/9mfq-cb36/rows.csv?accessType=DOWNLOAD"

covid_us <- read.csv(url_data)

head(corona_us, 10)
```

Let's clean the data a little bit

```r
covid_us <- covid_us %>%
  mutate(Date = submission_date) %>%
  mutate(Date2 = mdy(Date)) %>%
  separate(submission_date, sep = "/", into = c("month", "day", "year"))

# Sort the data in an increasing order
covid_us <- covid_us[order(covid_us$Date2), ]

head(covid_us)
```

As we can see there are a lot of **NA** values in the dataset, these NA values are because there were no cases reported in the US in the first two months of the 2020.

The data is still very broad, let's select data for the State of Minnesota.

```r
covid_mn <- subset(covid_us, state == "MN")

head(covid_mn)
```

Now we can explore the data in a similar way we did for the entire United States.

```r
plot(1:nrow(covid_mn), covid_mn$tot_cases, type = "b",
     ylab = "Infected", xlab = "Days since the pandemic started")
```

Modeling SIR

Now using the data from the US we will fit the SIR model to predict the changes in the number of Infected cases. The basic idea of the SIR model is, in fact, quite simple. First, we can see that the SIR model is composed of a set of three groups of people or compartments:

1. Those who are healthy but susceptible to the disease (**S**),
2. the infected (**I**),
3. the recovered (**R**).

Then, these compartments in turn are used to can model the dynamics of an outbreak. However, first of all we need to understand a little bit the *Math* behind the scenes. Jeannine provided a nice introduction to the math in her last lecture, but if you want to dig a little bit more you can watch this amazing video by Trevor Bazett The MATH of Epidemics | Intro to the SIR Model on YouTube. But, if you want to learn more details on this topic, the Book **Epidemics: models and data using R** can help you.

Anyhow, let's recap a little bit how the math works. To do that, we need three *differential equations*, one for the change in each group or compartment, where β (represent the infection rate) is the parameter that controls the transition between S and I and γ (gamma) which controls the transition between I and R, represents the removal or recovery rate.

The first equation indicates that the number of susceptible subjects (S) decreases with the number of newly infected subjects. In other words, every new infected subject is the result of the infection rate (β) times the number of susceptible individuals (S) who had a contact with an infected subject (I).

$$\frac{dS}{dt} = -\frac{\beta IS}{N}$$

The second equation indicates that the number of infected subjects (I) increases as new infected individuals are added to the pool (βIS) minus the recovered subjects (γI), where, γI is the removal rate γ times the infected subjects I.

$$\frac{dI}{dt} = \frac{\beta IS}{N} - \gamma I$$

Finally, the third equation indicates that the number of recovered subjects R increases with the number of subjects who either recovered or died (γI).

$$\frac{dR}{dt} = \gamma I$$

Okay, we are almost there. But first let's see how an epidemic develops.

- Step zero - An outbreak starts when no subject in the population has anti-bodies, in other words, when S (susceptible subjects) equal the entire population N.
- Step one - Once one subject in the population get infected (I), this subject change the state from S to I. Thus, S decreases by **1** as new I are added to the population.
- Step two - Each I (before recovering or dying) in turn infect other subjects in the population.
- Step three - The dynamic continues as newly I subjects infect other S subjects in the population before they recover (or died) or until some **Interventions** are applied (e.g., quarantine, wearing masks).
- Step four - Vaccines are developed (**YAY**), so a counteract can be applied to reduce the number of S or to increase the number of R.

As we now understand how the SIR model works, we can build an **R function** that allow us to model the dynamic of a disease.

```
SIR <- function(time, state, parameters) {
  par <- as.list(c(state, parameters))
  with(par, {
    dS <- -Beta * I * S / N # Equation one
    dI <- Beta * I * S / N - Gamma * I # Equation two
    dR <- Gamma * I # Equation three
    list(c(dS, dI, dR))
    })
}
```

Now to fit the SIR model to the data (US dataset) we need a ***solver*** and an ***optimizer***, why is that? Well, the SIR model is an ***Ordinary Differential Equation*** (**ODE**), thus, a solver function is needed to help us to solve the equations. We will use the function "**ode**" of the package {**deSolve**} and the "**optim**" function of {**R base**} as an optimizer.

More specifically, in order to solve the equations what we need is to minimize the sum of the squared differences (RSS) between the number of I subjects (infected) and time t–$I(t)$–and the number of predicted cases by our model, i.e., $\hat{I}(t)$:

$$RSS(\beta, \gamma) = \sum_{t} \left(I(t) - \hat{I}(t)\right)^2$$

Before start solving the SIR model, let's write another function that allows the RSS estimation.

```
RSS <- function(parameters) {
  names(parameters) <- c("Beta", "Gamma")
  out <- ode(y = init, times = Days, func = SIR, parms = parameters)
  # the out object includes the SIR function we wrote above
  fit <- out[, 3]
  sum((Infected - fit)^2)
}
```

Okay, we now have almost all information needed to fit our first SIR model. The last information we need is to set of initial values for N (population size), S, I and R.

By observing the trend in the number of cases for MN, we can see that the first record of infected subject was in March 06 2020, so, let's set that day as a starting point for our SIR model. Moreover, as more than a year has passed since the pandemic started, the variability in the number of cases in very huge, so let's select the first 30 days of cases to fit our model and then using the fitted model to predict for the next couple of months.

```
N <- 5686649 # Total population for the State of Minnesota for the 2020

start_date <- "2020-03-06"
end_date <- "2020-04-10"

# isolating the infected subjects in the state of Minnesota
Infected <- subset(covid_mn, Date2 >= ymd(start_date) & Date2 <= ymd(end_date))$tot_cases

Days <- 1:length(Infected) # Number of days since the first case
```

```
plot(Days, Infected, type = "b")
```

```
plot(Days, Infected, log = "y")
abline(lm(log10(Infected) ~ Days))
title("Confirmed Cases 2019-nCoV in MN, first 30 days", outer = TRUE, line = -2)
```

We can see that the confirmed cases in Minnesota were increasing rapidly since the first case of COVID-19 was reported.

As we now know that the number of I were increasing during the first days of the pandemic, let's model de dynamic of COVID-19 during that period of time.

```
init <- c(
  S = N - Infected[1], # Susceptible group
  I = Infected[1], # Infected group
  R = 0 # Recovered group
)
```

Now we can combine all information and run our model. Using the information provided above, we can find the values of β and γ that give the smallest RSS that represent the best fit to the data. Let's start exploring with values of **0.5** for each step and constrain those values to an interval from 0 to 1.

You can play with the initial values by changing the **c(0.5, 0.5)** to lower or higher values (e.g., 0.1 or 0.7), but remember, the estimates of β and γ will change. A nice post that explains in more detail the impact of changing the initial values can be found HERE.

```
Opt <- optim(c(0.5, 0.5),
  RSS,
  method = "L-BFGS-B",
  lower = c(0, 0),
  upper = c(1, 1)
)
```

Check if the model converged.

```
# optimize with some sensible conditions
Opt$message

# [1] "CONVERGENCE: REL_REDUCTION_OF_F <= FACTR*EPSMCH"
```

From the optimized model we can obtain the β and γ values, and remember, these values controls for the transition between S (susceptible) and I (infectious) and I and R (recovered), respectively.

```
Opt_par <- setNames(Opt$par, c("Beta", "Gamma"))
Opt_par
```

Now using the fitted β and γ values, we can see if our model recovers the observed trend in the US.

```
# get the fitted values from our SIR model
fit_incidence <- data.frame(ode(
  y = init, times = Days,
  func = SIR, parms = Opt_par
))

head(fit_incidence)
tail(fit_incidence)
```

Let's plot the estimated I over the 30 days. We will use a kind of plot named **semi-log plot**, this plot allow us the read the difference between the expected (fitted) and observed number of confirmed cases over time.

```
matplot(fit_incidence$time, fit_incidence$I,
        type = "l", log = "y",
        xlab = "Days", ylab = "Number of infected subjects",
        lwd = 2, lty = 1)
points(Days, Infected)
```

Hmmm, looks like the fitted model (despite it converged well) does not accurately recover the number of infected cases in the Minnesota, in fact, it tend to underestimate the number of cases, although we can see an steady increment on the number of cases since the first detected case.

How do you feel about that? Do you see any other interested pattern? If so, please explain

Reproduction number R_0 (R naught)

Although our SIR model does not fit the observed data accurately, we can still use it to estimate the basic reproduction number R_0 (AKA reproductive ratio). The R_0 give us an approximate estimation of how many S subjects get infected by a sick subject (I) on average. To calculate R_0 we just need to obtain the ratio between β and γ:

$$R_0 = \frac{\beta}{\gamma}$$

```r
R0 <- setNames(Opt_par["Beta"] / Opt_par["Gamma"], "R0")
round(R0, 3)
```

The estimated R_0 is 1.549 which suggests that for every sick subject 1.5 subjects can get infected by COVID-19. This value is lower when compared to other diseases and even lower than the estimated R_0 for COVID-19 in the literature that ranges between 2.5 and 3. One potential explanation for this low R_0 can be due the fact the first 30 days, the number of observed cases were very low when compared with the large population size (N) in the state of Minnesota.

Evaluating the outbreak under no intervention

To fit our model we used the first 30 days since the first reported case, now let's explore what would happen if no public intervention (i.e., quarantine) was applied. Here, using the fitted model we will extrapolate up to 90 days

```r
times <- 1:90 # time in days

fit_90 <- data.frame(ode(
  y = init, times = times,
  func = SIR, parms = Opt_par))

head(fit_90)
tail(fit_90)
```

To better explore the data, let's make some figures that allow us to see what happened the first 90 days of the pandemic in the state of Minnesota.

```r
cols <- 1:3 # colors: black = susceptible, red = infected and green = recovered
```

```r
matplot(fit_90$time, fit_90[, 2:4], type = "l",
        xlab = "Days", ylab = "Number of subjects",
        lwd = 2, lty = 1, col = cols)
legend("left", c("Susceptible", "Infected", "Recovered"),
       lty = 1, lwd = 2, col = cols, inset = 0.05)
```

Same figure but in log scale and adding the observed cases

```r
matplot(fit_90$time, fit_90[ , 2:4], type = "l",
        xlab = "Days", ylab = "Number of subjects",
        lwd = 2, lty = 1, col = cols, log = "y")
## Warning in xy.coords(x, y, xlabel, ylabel, log = log): 1 y value <= 0
## omitted from logarithmic plot
```

```
points(Days, Infected)
legend("bottomright", c("Susceptible", "Infected", "Recovered"),
       lty = 1, lwd = 2, col = cols, inset = 0.05)
title("SIR model 2019-nCoV United States", outer = TRUE, line = -2)
```

How do you feel about that? Please explain the trend of each compartment in a temporal way?

Additional summary statistics

Using the fitted model we can also estimate some additional statistics that could help us to better understand the dynamics of a pandemic:

- Peak of the pandemic
- Fatality rate

Peak of the pandemic

```
# Peak of the pandemic for the first 60 days
fit_incidence[fit_incidence$I == max(fit_incidence$I), c("time", "I")]
```

Fatality rate

```
max(fit_incidence$I) * 0.02 # Assuming 2% of fatality rate
```

The challenge

The challenge for this assignment is to repeat the process by selecting other state in the United States and discuss if the R_0 obtained differ to the obtained for the state of Minnesota.

References

Bjørnstad, O. N. (2018). Epidemics. Use R! doi:10.1007/978-3-319-97487-3

That's all!