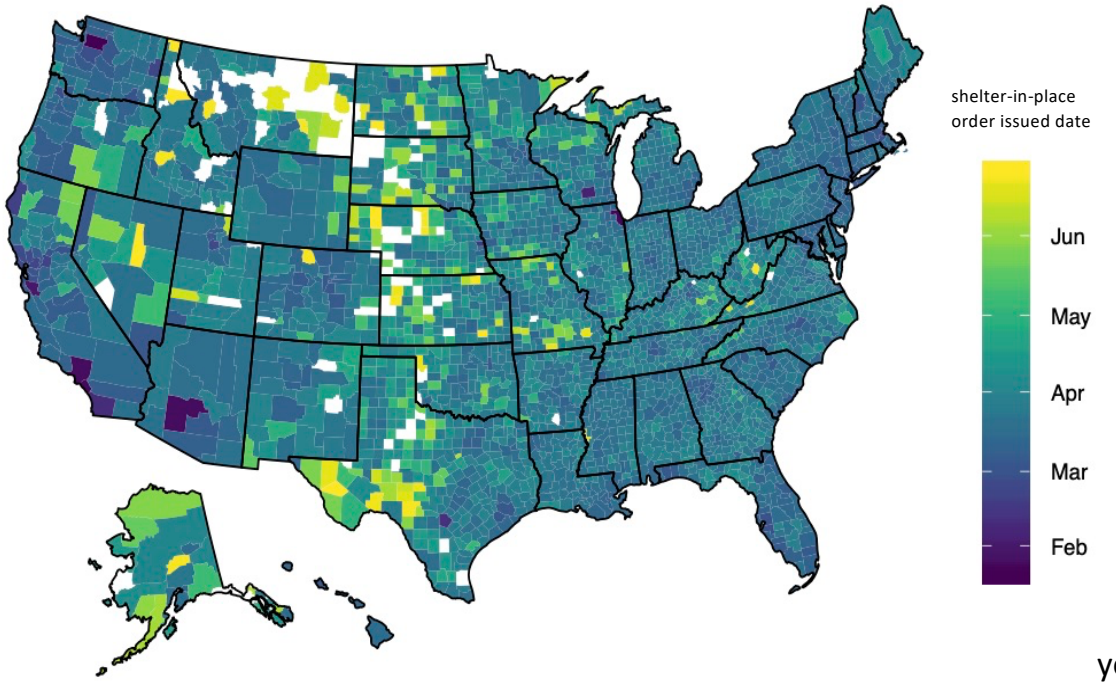
S&DS 177
YData: COVID-19 Behavioral Impacts



Youpei Yan
youpei.yan@yale.edu

Lecture 2 (Feb 11, Thursday 9:25 – 11:15)

Overview:

- SIR Model
- Practice of SIR Model in Jupyter
- Importing data or Downloading data
- Practice the simple COVID-19 data selecting & sorting

Last week:

- Introduction of the course website
- Course overview
- Discussion of the impact of pandemic
- Jupyter Notebook (Lab 0 discussion)

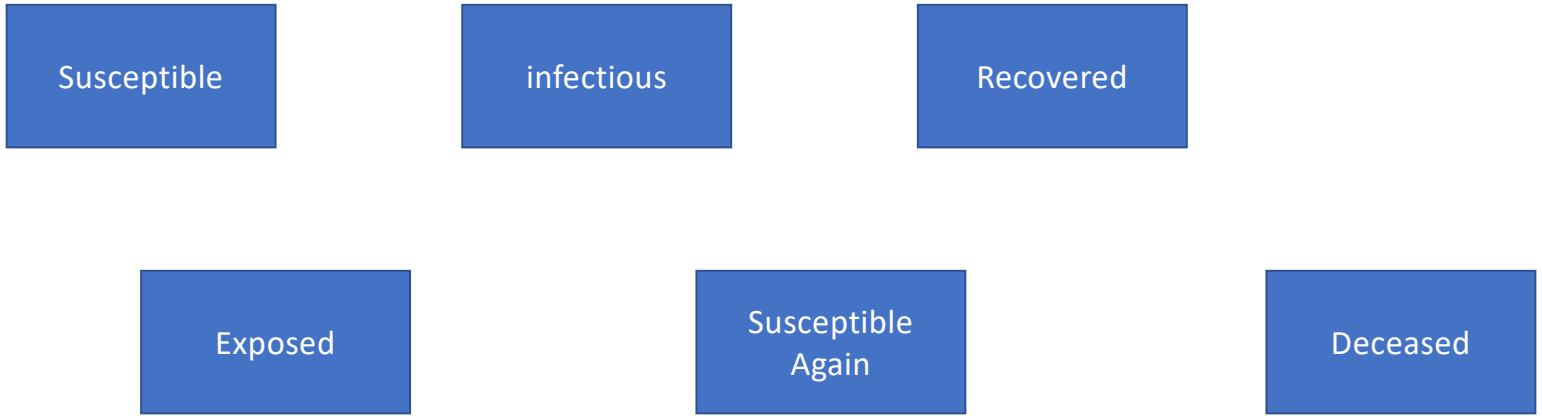
Lecture 2 (Feb 11, Thursday 9:25 – 11:15)

- Compartmental models
  - Simplified mathematical modelling of infectious diseases.

S-I-R Model

Lecture 2 (Feb 11, Thursday 9:25 – 11:15)

- Compartmental models
  - Simplified mathematical modelling of infectious diseases.



SIS; SEIS; SIRD; etc.

Lecture 2 (Feb 11, Thursday 9:25 – 11:15)

Compartmental models

- Simplified mathematical modelling of infectious diseases.
- **S**: # **s**usceptible individuals.
    - When a susceptible and an infectious individual have an "infectious contact"
    - the **S** individual contracts the disease and moves to the **I** compartment.
- **I**: # **i**nfectious individuals.
    - Already infected and are capable of infecting susceptible individuals.
- **R**: # **r**ecovered (and immune) individuals.
    - Infected but recovered from the disease and entered the removed compartment (or died).
    - Given the total population pool is huge, it is assumed that the number of deaths is negligible.

Lecture 2 (Feb 11, Thursday 9:25 – 11:15)

- Compartmental models

  - N: total population.
  - β : infection-producing contacts per unit time (The contact that will effectively infect another person)

  An infected individual comes into contact with βN other individuals per unit time, and get βn infected.

  - n: the fraction of the population that are susceptible to contracting the disease.
  - $n = S/N$.

  - γ : mean recovery rate.
  - $1/\gamma$: mean period of time during which an infected individual can pass it on.

Lecture 2 (Feb 11, Thursday 9:25 – 11:15)

- Compartmental models

  - N: total population.
  - β : infection-producing contacts per unit time (The contact that will effectively infect another person)

  An infected individual comes into contact with βN other individuals per unit time, and get βn infected.

  - n: the fraction of the population that are susceptible to contracting the disease.
  - $n = S/N$.

The change of susceptible # : $$\frac{\mathrm{d}S}{\mathrm{d}t} = -\frac{\beta SI}{N}$$

Lecture 2 (Feb 11, Thursday 9:25 – 11:15)

- Compartmental models

  - N: total population.
  - β : infection-producing contacts per unit time (The contact that will effectively infect another person)

  An infected individual comes into contact with βN other individuals per unit time, and get βn infected.

  - n: the fraction of the population that are susceptible to contracting the disease.
  - $n = S/N$.

Ex: 1 infected person, with 25% rate of infection, go out 1 day in a week, and meet 10 people. How many people will be affected per day?
0.25*1/7*10 = 0.3571.

This one person ”infect ~0.3571 people every day.”

Lecture 2 (Feb 11, Thursday 9:25 – 11:15)

The change of infectious # :

The change of susceptible # : $\frac{\mathrm{d}S}{\mathrm{d}t} = -\frac{\beta SI}{N}$

The change of recovered # :

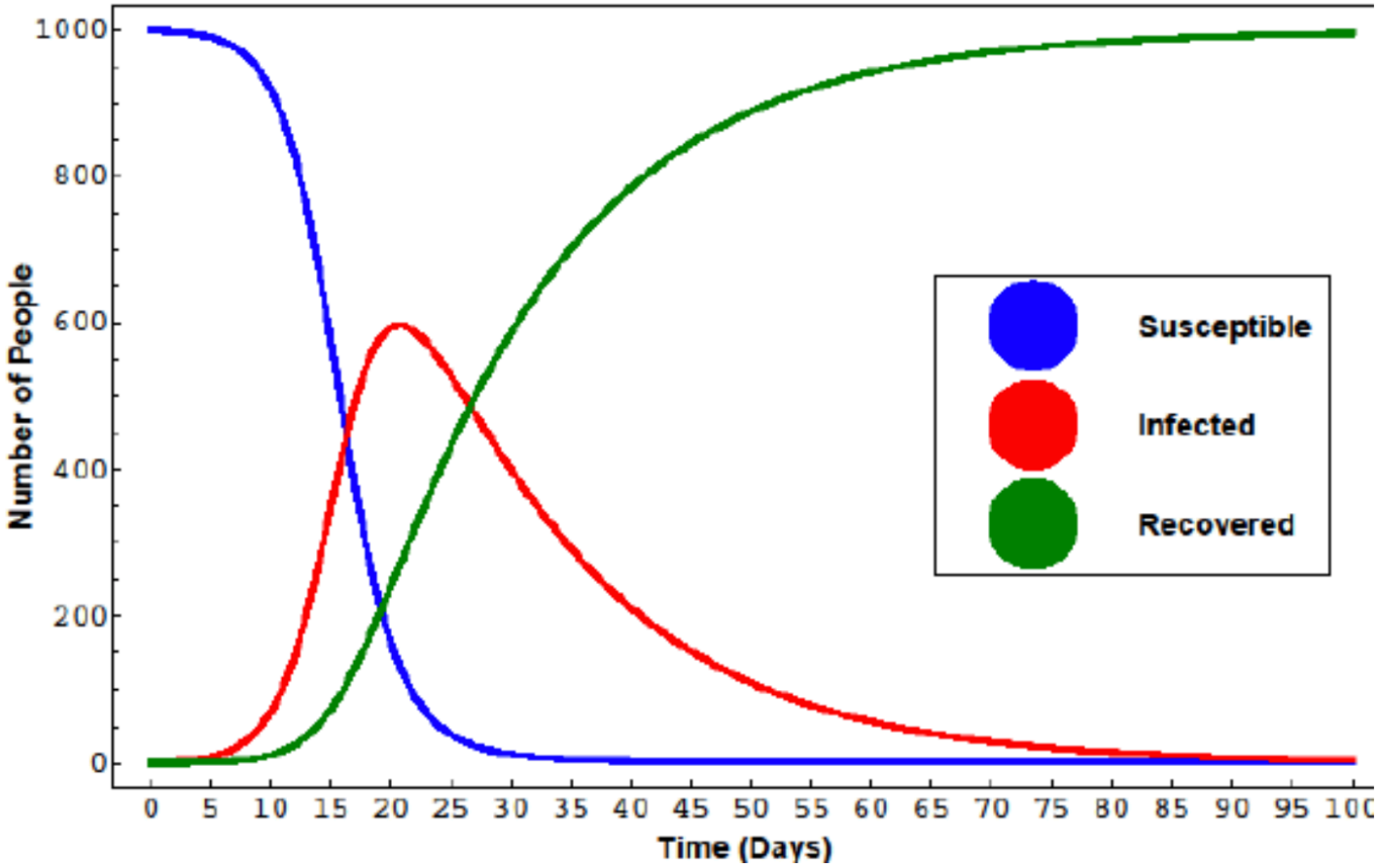
- γ : mean recovery rate.
- $1/\gamma$: mean period of time during which an infected individual can pass it on.

Ex: 1 infected person, with 14 days to recover.

This one person ”remove 1/14 people away from the infected pool every day.”

R0: Average new infections generated by each infected people.

Lecture 2 (Feb 11, Thursday 9:25 – 11:15)



Credit: https://www.researchgate.net/publication/224209140_To_agent-based_simulation_from_System_Dynamics

Lecture 2 (Feb 11, Thursday 9:25 – 11:15)

- How to do it in Python?

  - Set up the parameters (or use the real-world data to estimate).
  - Write a function with differential equations.
  - Solve the differential equations with the built-in function **odeint**.
  - Plot.

Lecture 2 (Feb 11, Thursday 9:25 – 11:15)

- How to do it in Python?

odeint:

odeint requires three inputs:
y = odeint(model, y0, t)

model: Function name that returns derivative values at requested y and t values as
dydt = model(y,t)

y0: Initial conditions of the differential states

t: Time points at which the solution should be reported. Additional internal points are often calculated to maintain accuracy of the solution but are not reported.

Lecture 2 (Feb 11, Thursday 9:25 – 11:15)

- Issue of SIR under the pandemic.
    - Virus has a relatively long incubation period.
        - People infect others before their own symptoms develop.
        - Their infectiousness appears to be significantly lower than that of individuals who are in different stages of developing symptoms, but may still significantly contribute to the epidemic dynamics.
    - Significant age differences in symptom development and prognosis.
    - No long-term immunity, people can be reinfected from COVID-19.

- Possible solution, S-L-A-P-I-R-D model? (https://www.nature.com/articles/s41598-020-77628-4)
    - L: Latent (who have contracted the virus but are not yet contagious);
    - A: Asymptomatic individuals (who have contracted the virus, may infect others, but will never present any symptoms);
    - P: Pre-symptomatic individuals (who have contracted the virus, are able to infect others, and have not yet, but will soon develop symptoms)

Lecture 2 (Feb 11, Thursday 9:25 – 11:15)

Last time:

1. Jupyter Notebook
   - 1.1 Text Cells
   - 1.2 Code Cells
   - 1.3 Writing Jupyter Notebook
   - 1.4 Errors
   - 1.5 The Kernel
2. Practice SIR
3. Import an online Dataset
4. Import a Local Dataset
   - 2.1 Loading data tables
   - 2.2 Selecting a subset of columns from a table
   - 2.3 Selecting a subset of rows from a table
   - 2.4 Using methods on values in a column of data in a table
   - 2.5 Sorting values in a column of data in a table

```
Table.read_table('url')
my_table.num_columns
my_table.num_rows

my_table.select('col_name_1', 'col_name_2', 'col_name3')
my_table.select(0,1,5)

my_table.take(2,5,6)

my_selected_column.sum()
my_selected_column.mean()
my_selected_column.max()

my_table.sort('col_name')

read_url('online_url')
pd.read_csv('online_url',sep=",")
```

Lecture 2 (Feb 11, Thursday 9:25 – 11:15)

Finally:

Way to save the file in both .pdf and .ipynb format (so any changes we made here can be saved and printed)

To produce the .pdf, please do the following in order to preserve the cell structure of the notebook:

1. Go to "File" at the top-left of your Jupyter Notebook
2. Under "Download as", select "HTML (.html)"
3. After the .html has downloaded, open it and then select "File" and "Print" (note you will not actually be printing)
4. From the print window, select the option to save as a .pdf

To produce the .ipynb, please do the following:

1. Go to "File" at the top-left of your Jupyter Notebook
2. Under "Download as", select "Notebook (.ipynb)"