

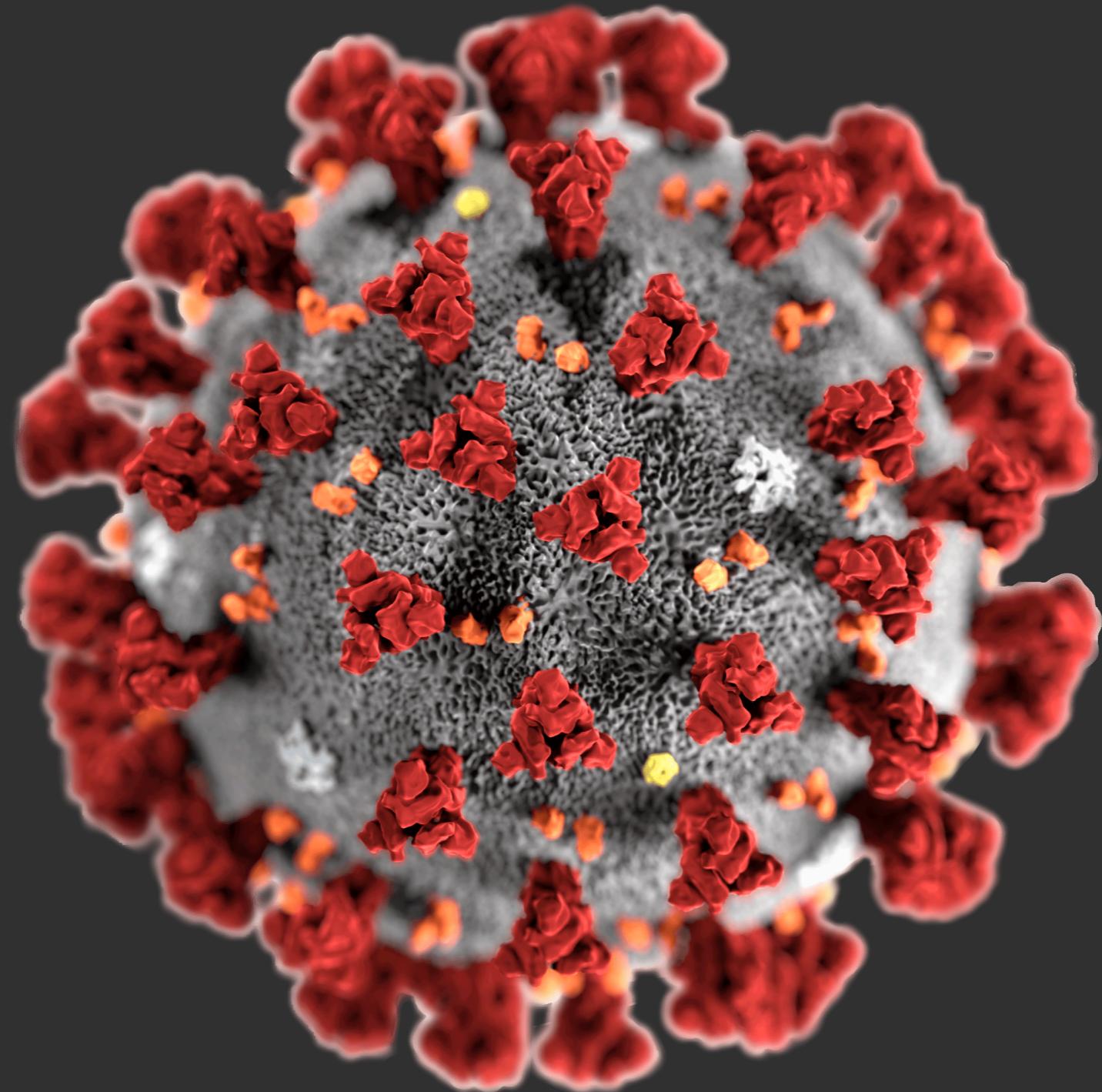


Understanding COVID-19 with Pandas and NetworkX

Bruno Gonçalves

www.data4sci.com/newsletter

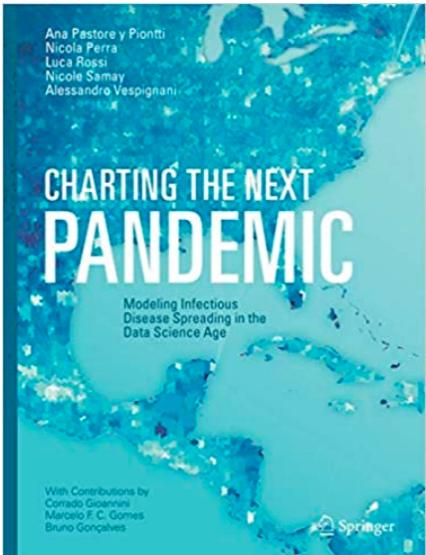
<https://github.com/DataForScience/CoVID19>



Who Am I?



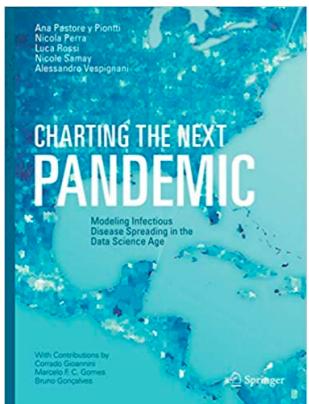
- **Name:** Bruno Gonçalves
- **Work:** Research Data Scientist
- **Background:** Physics (PhD) and Computer Science (MS)
- **Experience:** 10+ years using large scale datasets to analyze individual human behavior.
- **Website:** www.data4sci.com
- **Twitter:** @data4sci
- **Email:** bgoncalves@data4sci.com



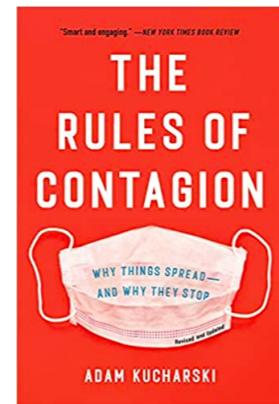
<https://amzn.to/3owsbZn>

References

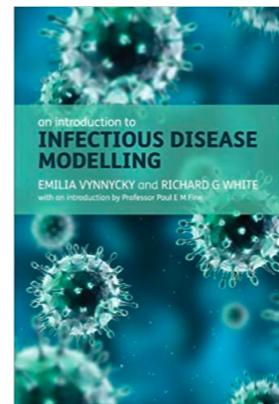
bgoncalves.medium.com/list/epidemiology-101-185b95a0f345



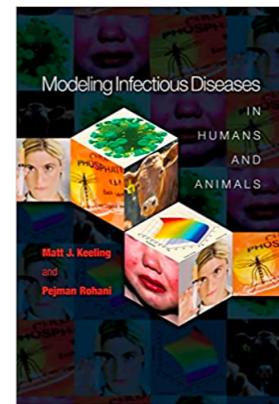
<https://amzn.to/3owsbZn>



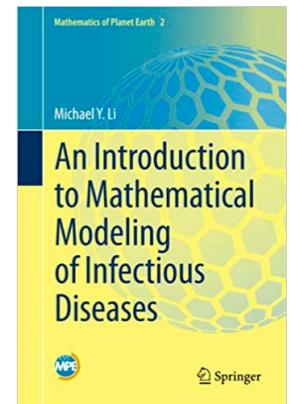
<https://amzn.to/3rxeift>



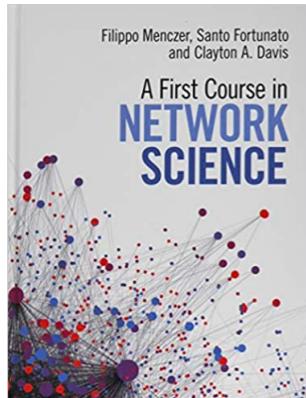
<https://amzn.to/3J8lWII>



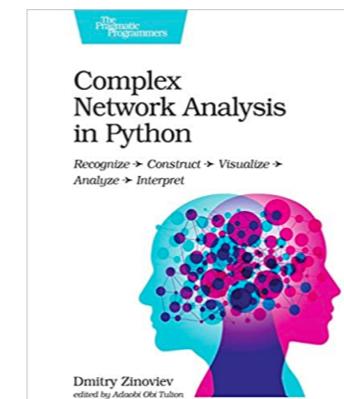
<https://amzn.to/3HDeteU>



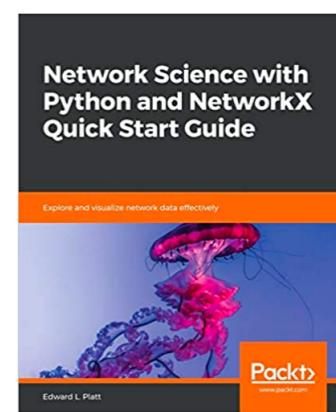
<https://amzn.to/3skkuH6>



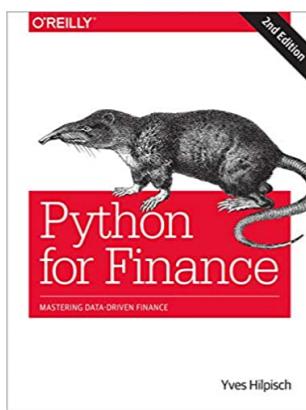
<https://amzn.to/3B5GuJN>



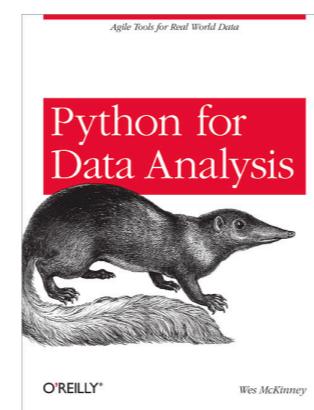
<https://amzn.to/3GA55XY>



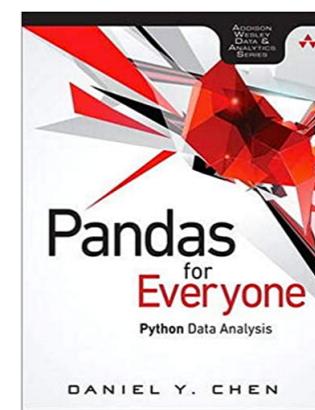
<https://amzn.to/3Lgxx4U>



<https://amzn.to/3bnycQu>



<https://amzn.to/2Pmg6UX>



<https://amzn.to/3bk8fkV>



Table of Contents

1. Background Information
2. Compartmental Models
3. Stochastic Models
4. Network Models
5. Vaccination



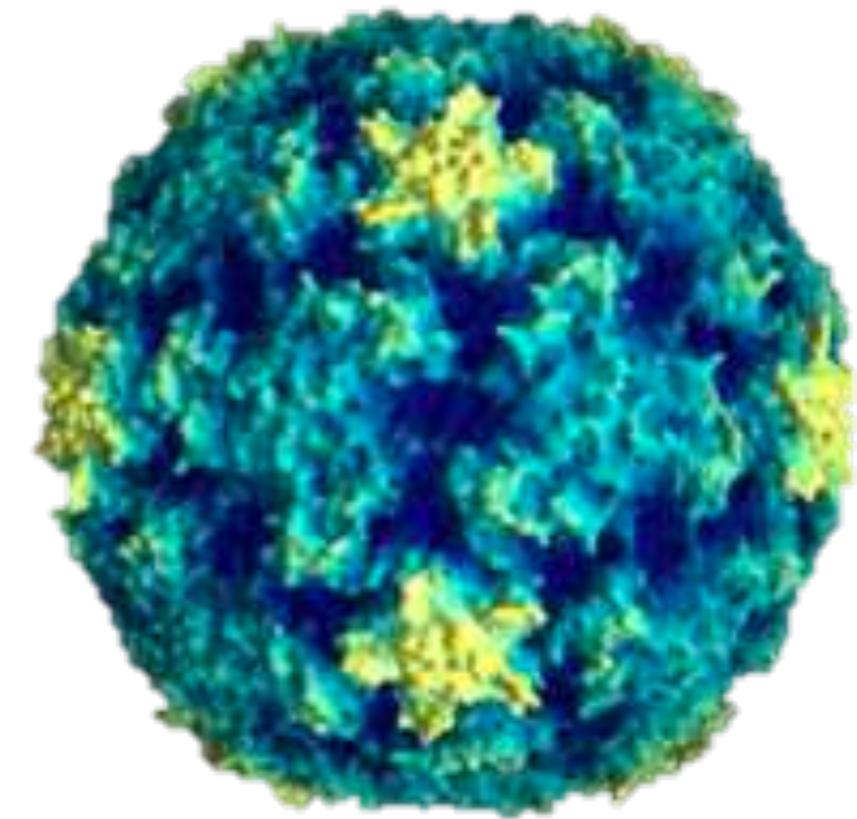
1. Background Information

Naming Games

- **Coronavirus** — A family of **RNA** viruses that cause respiratory tract infections in mammals and birds. Some of the viruses that cause the **Common Cold** are part of this family.
- **SARS** — *Severe Acute Respiratory Syndrome* — Caused the 2002–2004 SARS epidemic in Southeast Asia. Animal origin virus that arose in bats and is known as **SARS-CoV-1**.
- **MERS** — *Middle East Respiratory Syndrome* — Close cousin of the **SARS-CoV-1**, the **MERS-CoV** virus that originated in bats and spread to humans through camels.
- **CoVID-19** — *CoronaVirus Infectious Disease-2019* — Caused by the SARS-CoV-2 virus, another close cousin of the SARS virus that originated in bats and it thought to have spread to humans through pangolins.
- **2019-nCoV** — *2019 Novel Coronavirus* — The original name of SARS-CoV-2 before its close connection with SARS became clear.
- **Influenza** — A family of virus that causes the season flu. Many different viruses (**A/H1N1**, **A/H2N2**, **A/H3N2**, **A/H5N1**, B, C, etc) that mutate rapidly and circulate widely in the entire human population. Zoonotic origin, that arise in a variety of animals (birds, swine, etc) before spreading to humans. The last major flu pandemic was the **2009 A/H1N1pdm** pandemic, known as the Swine Flu pandemic. In the US, the CDC estimates that in the 2017–2018 season the flu infected 45,000,000 Americans, resulting in 61,000 deaths.

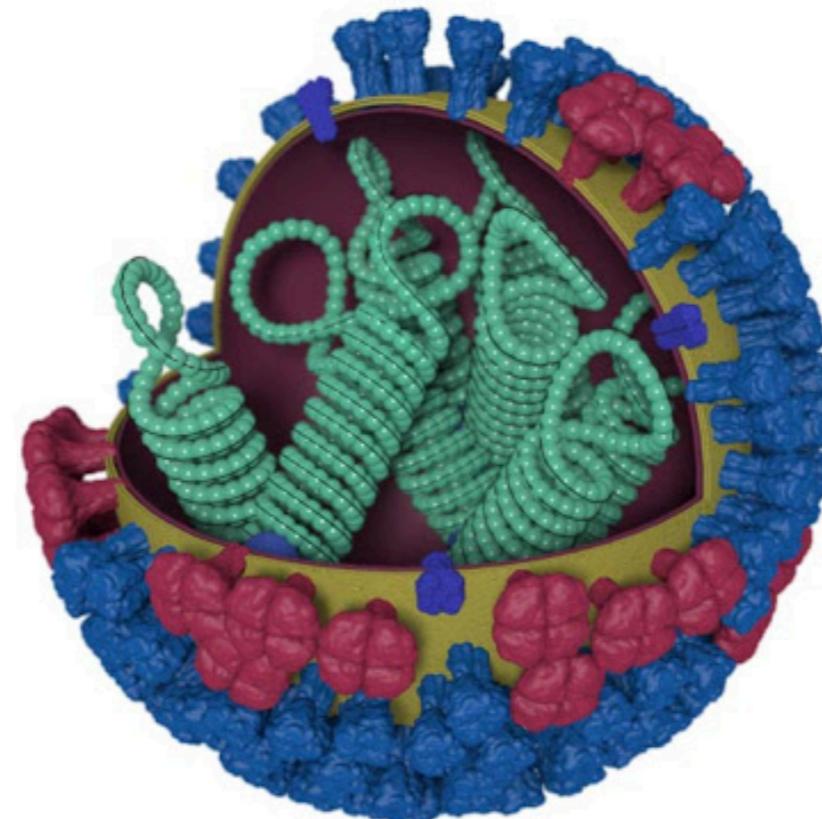
Common Viruses

Rhinovirus



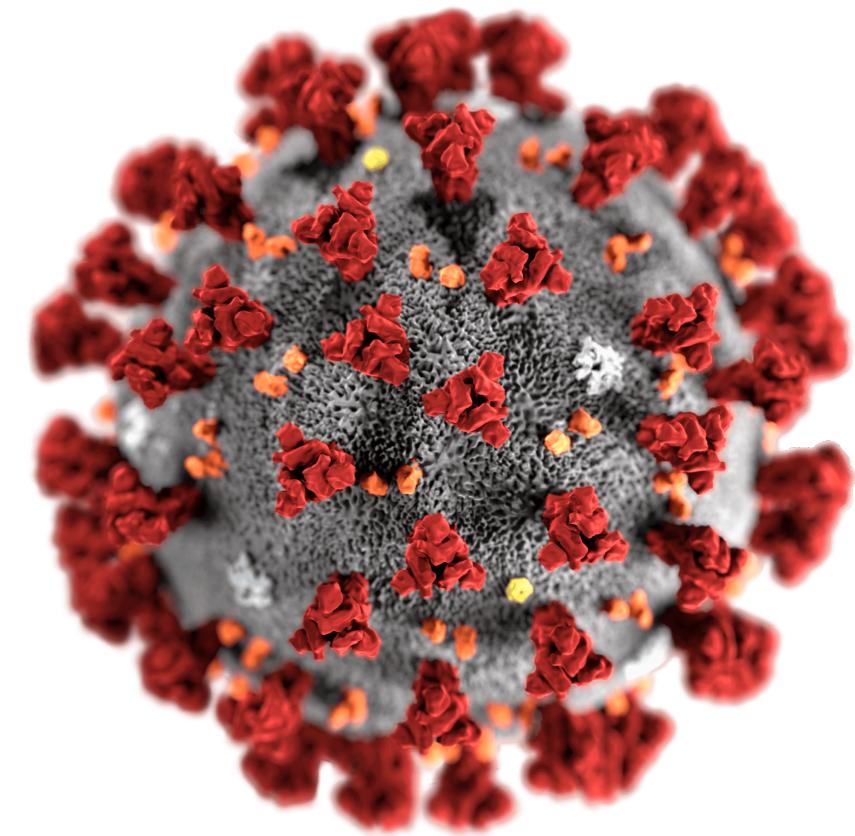
Common Cold

Influenza



Flu

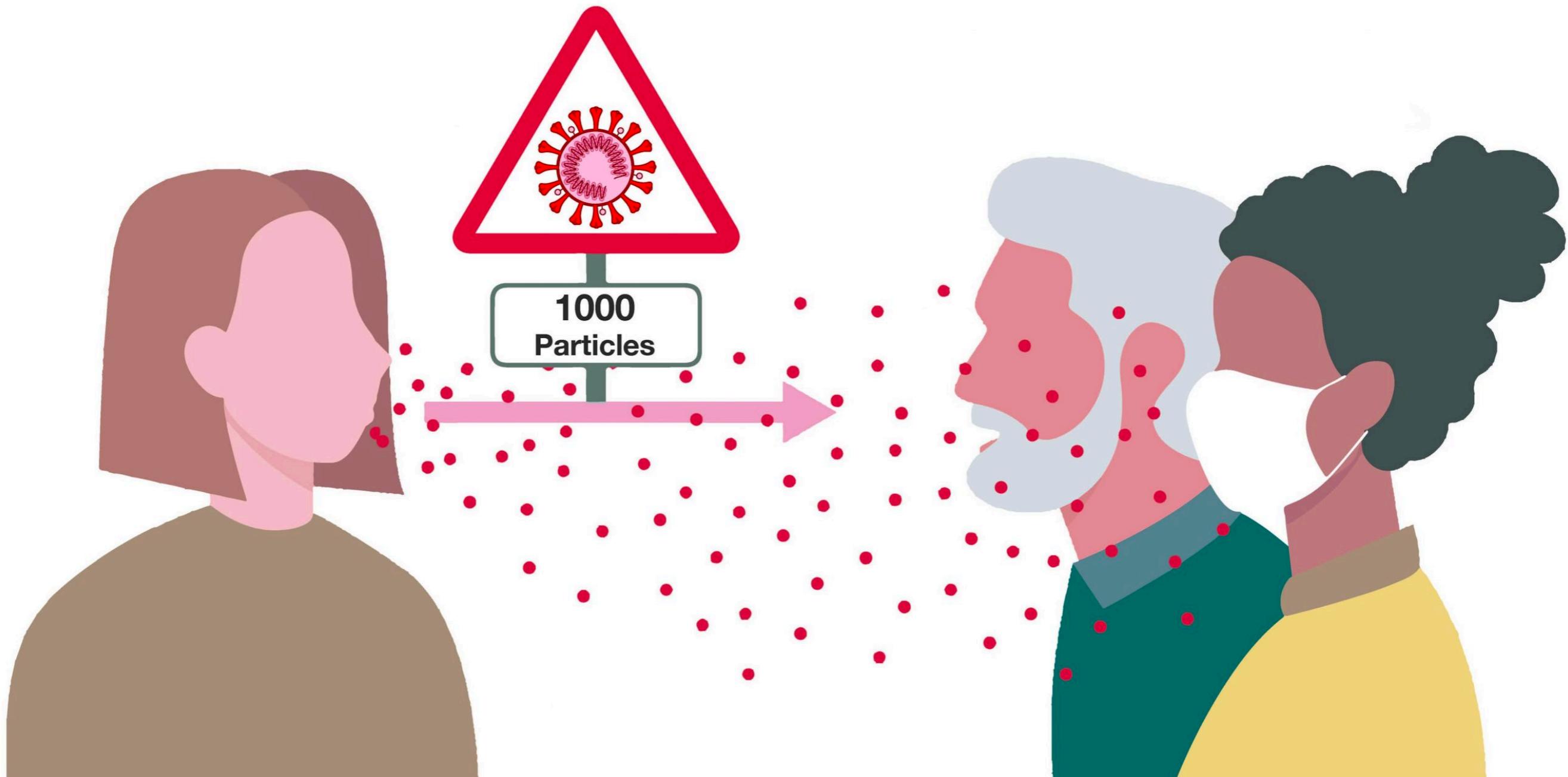
Coronavirus



SARS-CoV-2

- Viruses are essentially little bags of genetic material.
- Two “flavors” - RNA and DNA
- Different viruses transmit through different mechanisms and pathways
- Cold/Flu viruses transmit through the respiratory tract

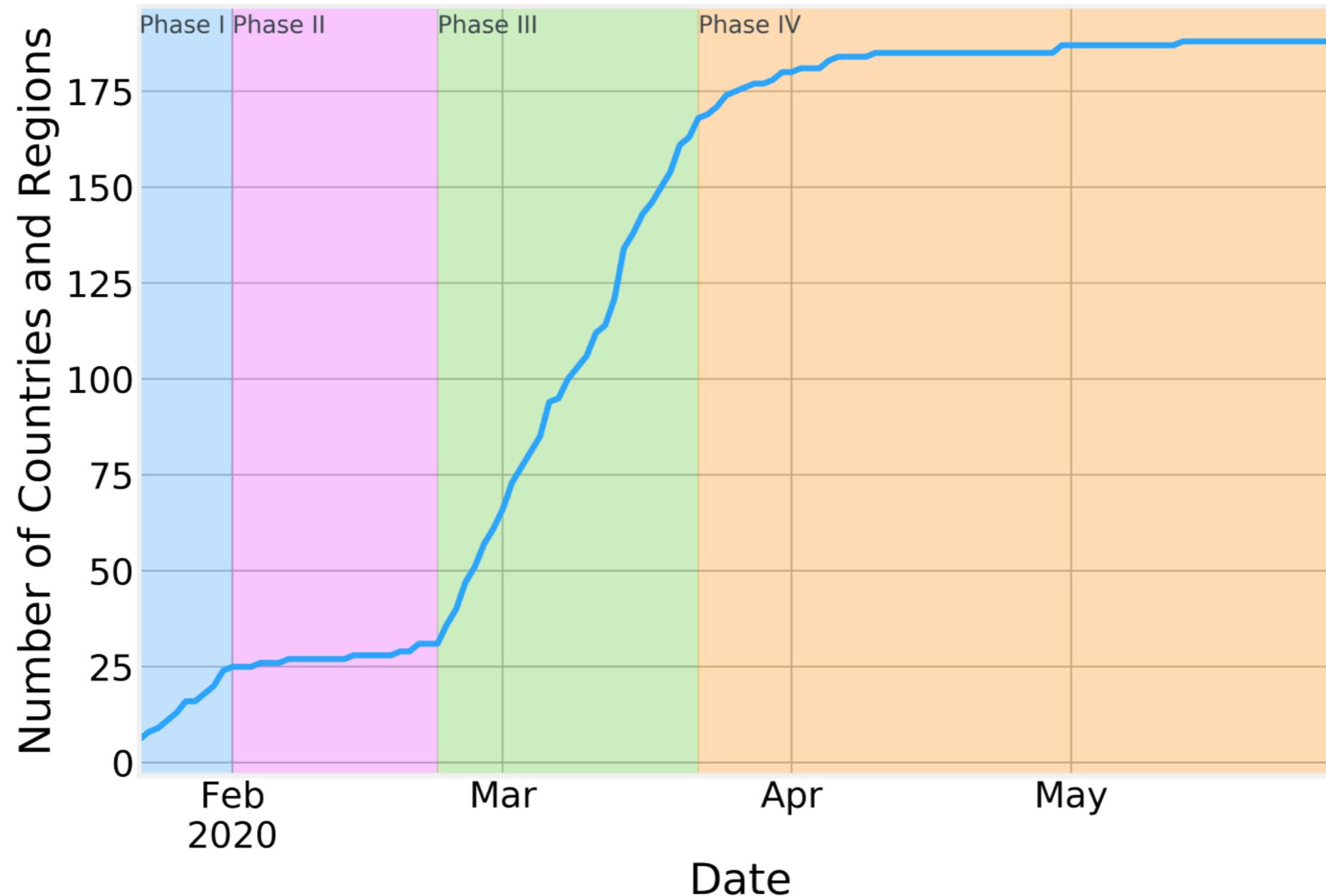
ILI Transmission



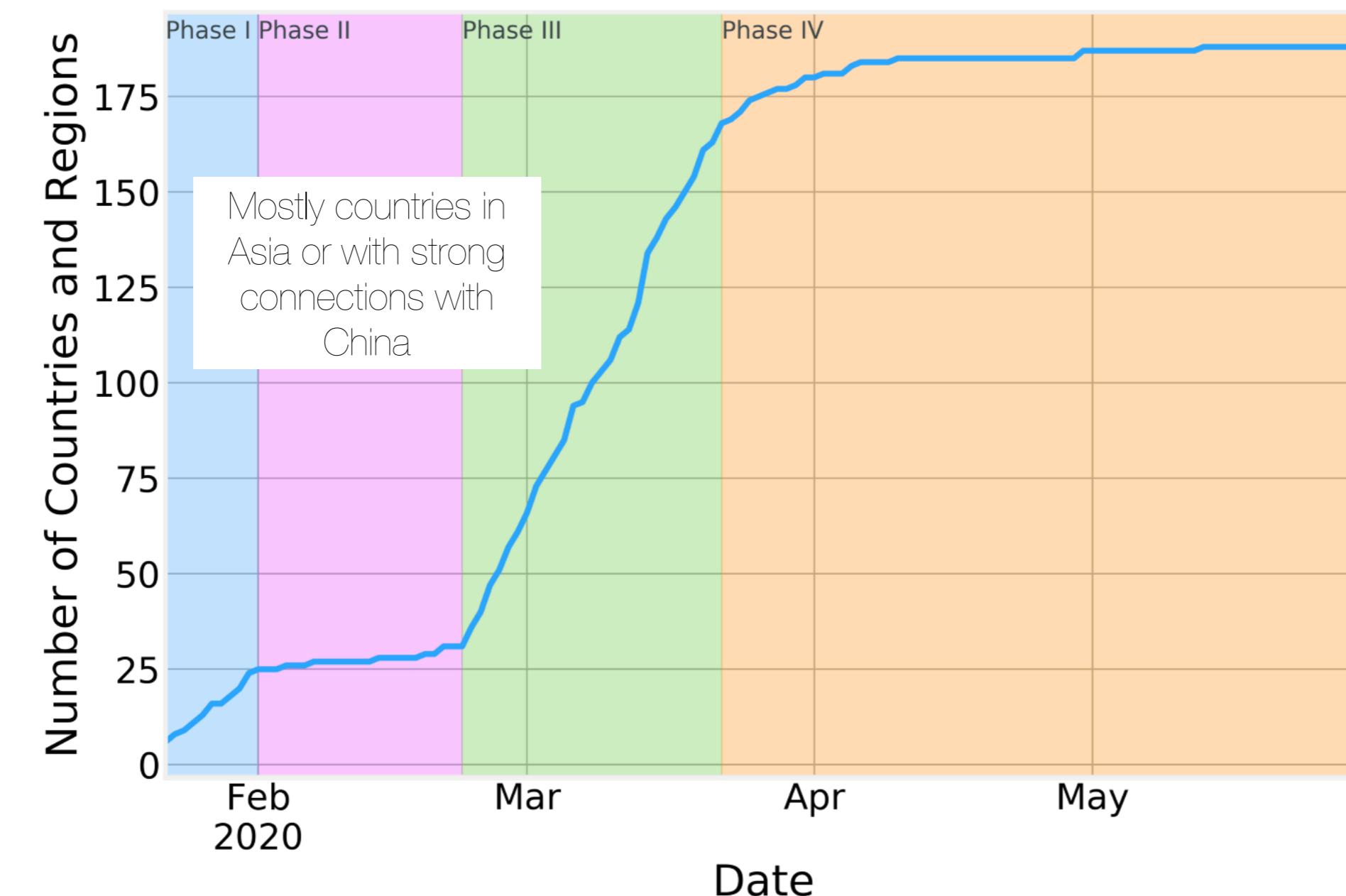
Early Timeline

- Dec 1 - First patient presents symptoms in Wuhan, China
- Dec 18 - Total of 9 cases detected, 2 of which linked to the Huanan Seafood Wholesale Market
- Dec 21 - Cluster of patients with "pneumonia of an unknown cause" had been identified
- Dec 29 - 4 patients from Huanan Seafood Wholesale Market admitted to hospital with pneumonia
- Dec 30 - [Dr. Li Wenliang](#), posted on [WeChat](#) that a cluster of 7 patients had been diagnosed with SARS based on RNA analysis.
- Dec 31 - Various hospitals in Wuhan held an emergency symposium on the topic of the treatment of pneumonia patients of unknown cause which had been diagnosed in various Wuhan, China, hospitals.
- Jan 1 - Hunan Seafood Market closed for renovations.
- Jan 7 - [CDC](#) announces travel advisory
- Jan 8 - New Coronavirus discovery announced
- Jan 19 - First cases outside Wuhan
- Jan 23 - Quarantine of "[greater Wuhan](#)" announced
- Jan 29 - Companies start to announces 14 day "work from home" for anyone returning from China
- Feb 2 - [US](#) announces [14 day](#) quarantine for returning travelers
- Feb 9 - Number of fatalities [surpasses](#) the total number for [SARS](#)

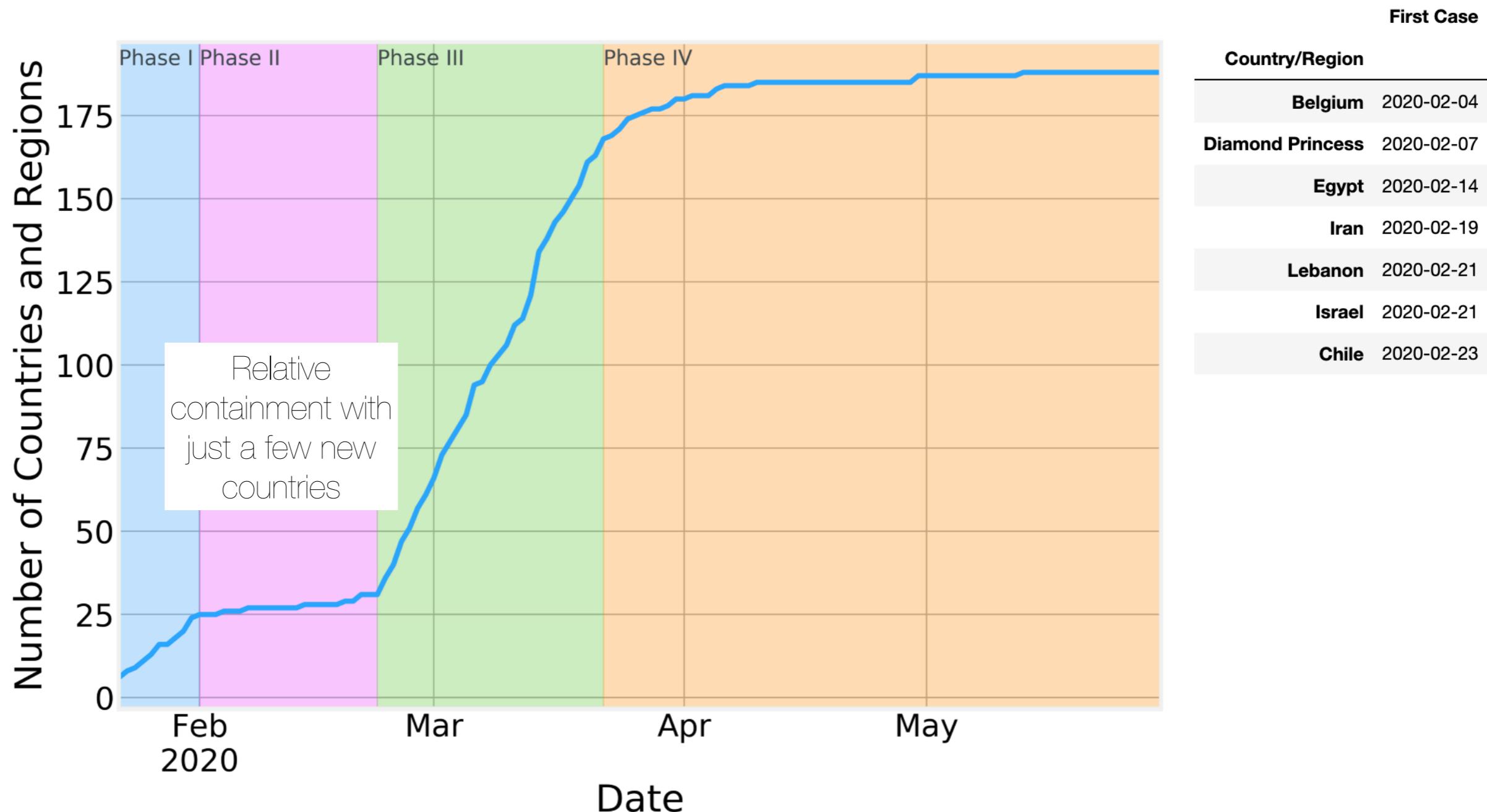
Global Spread



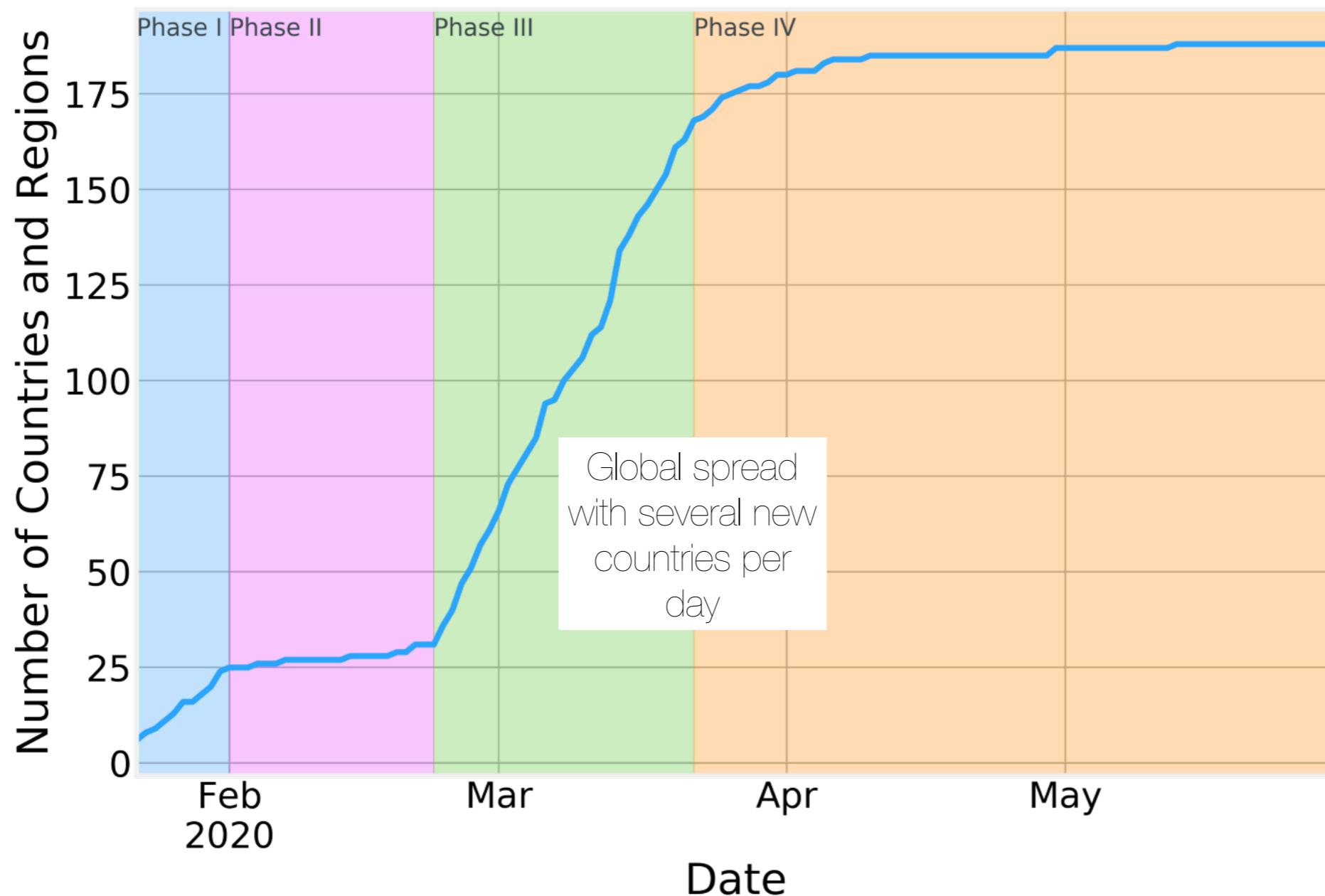
Global Spread - Phase I



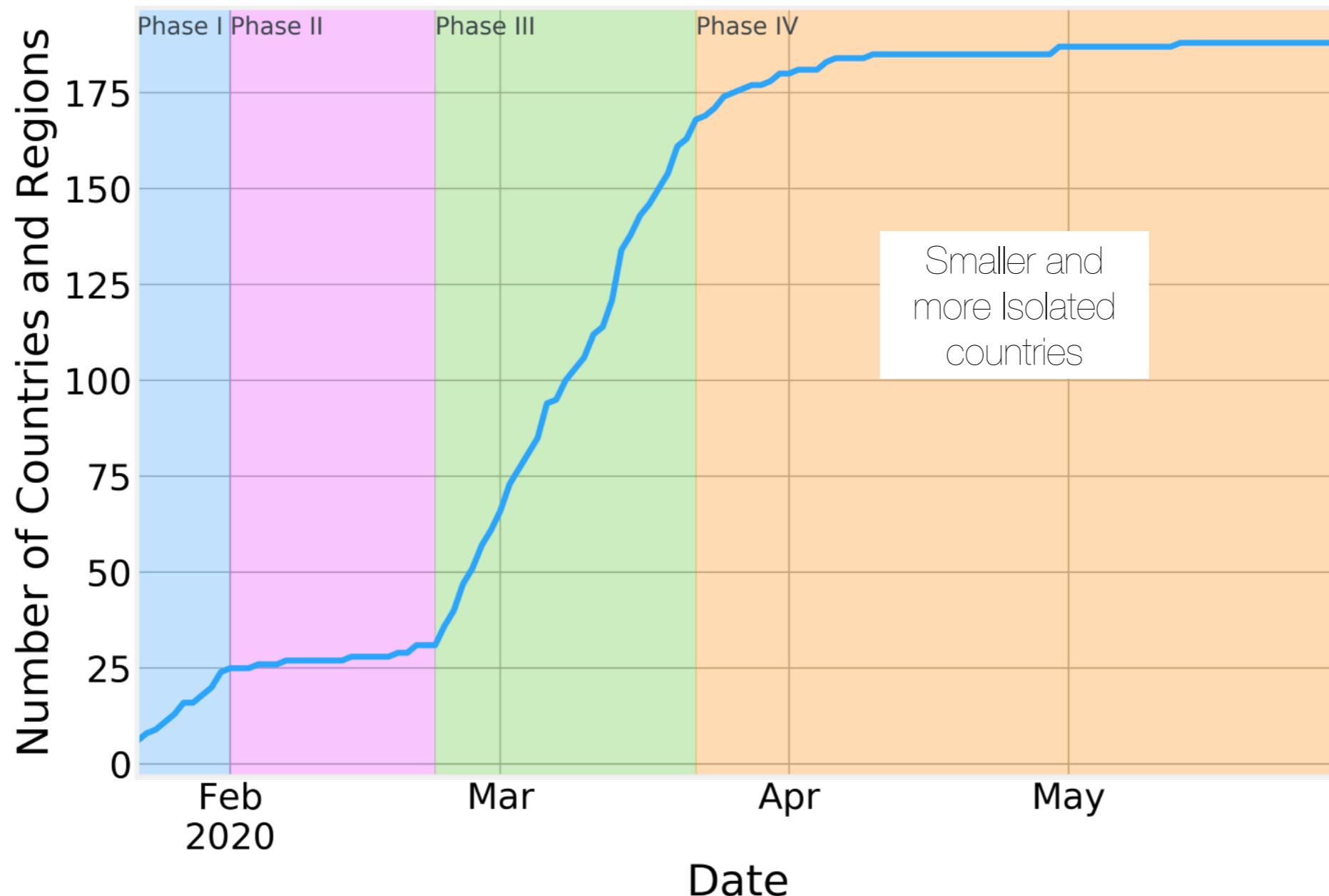
Global Spread - Phase II



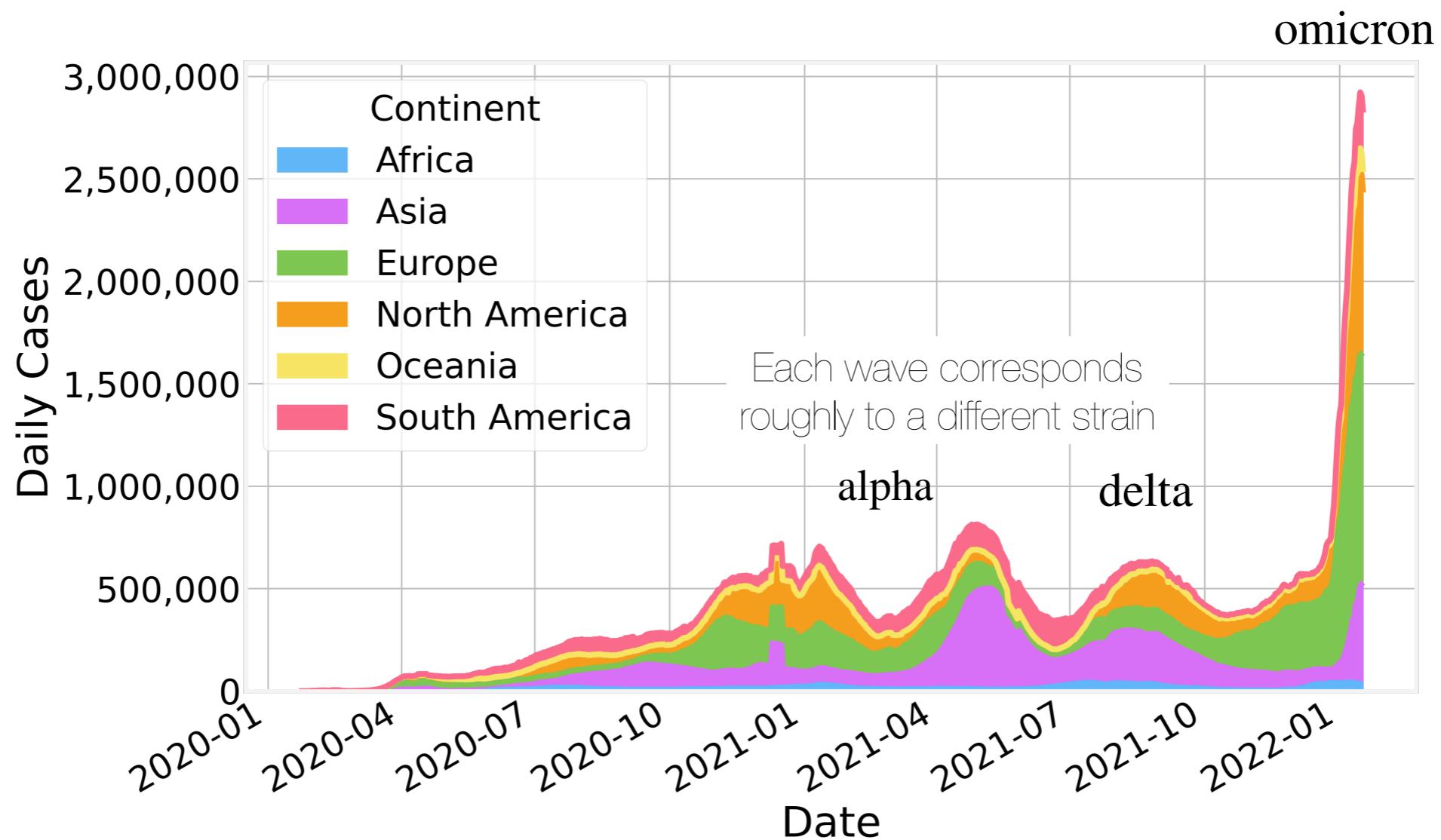
Global Spread - Phase III



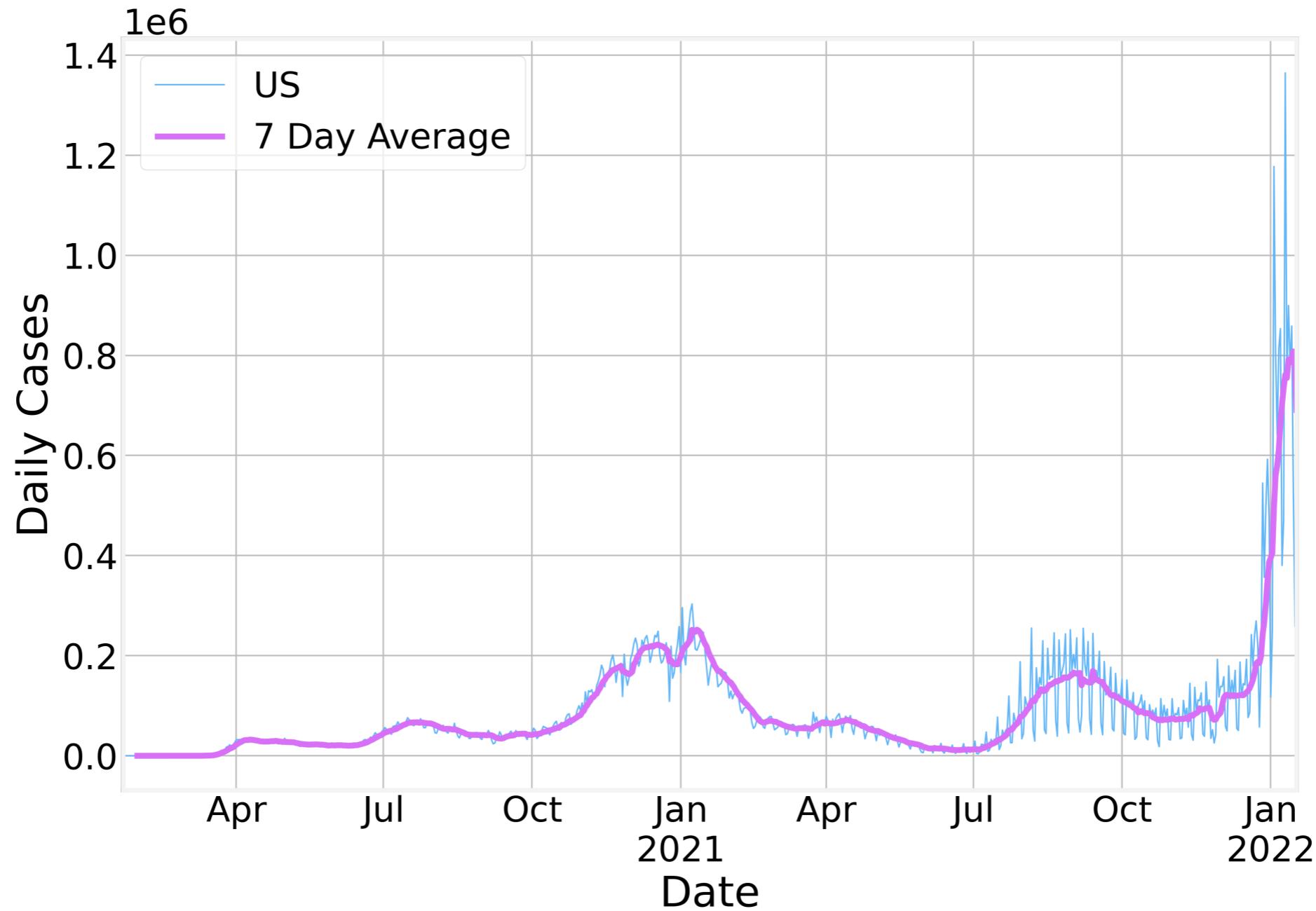
Global Spread - Phase IV



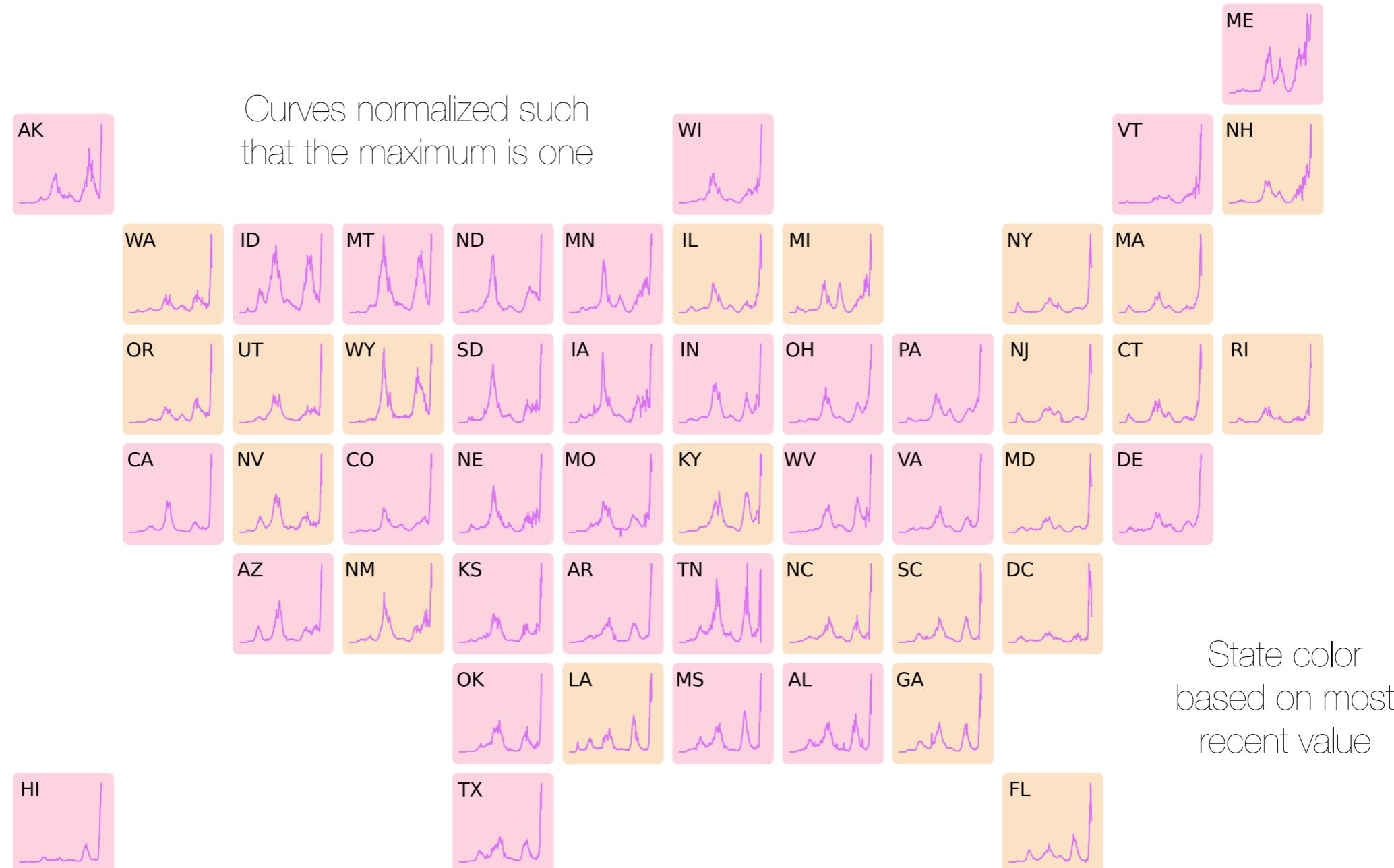
Global Number of Cases



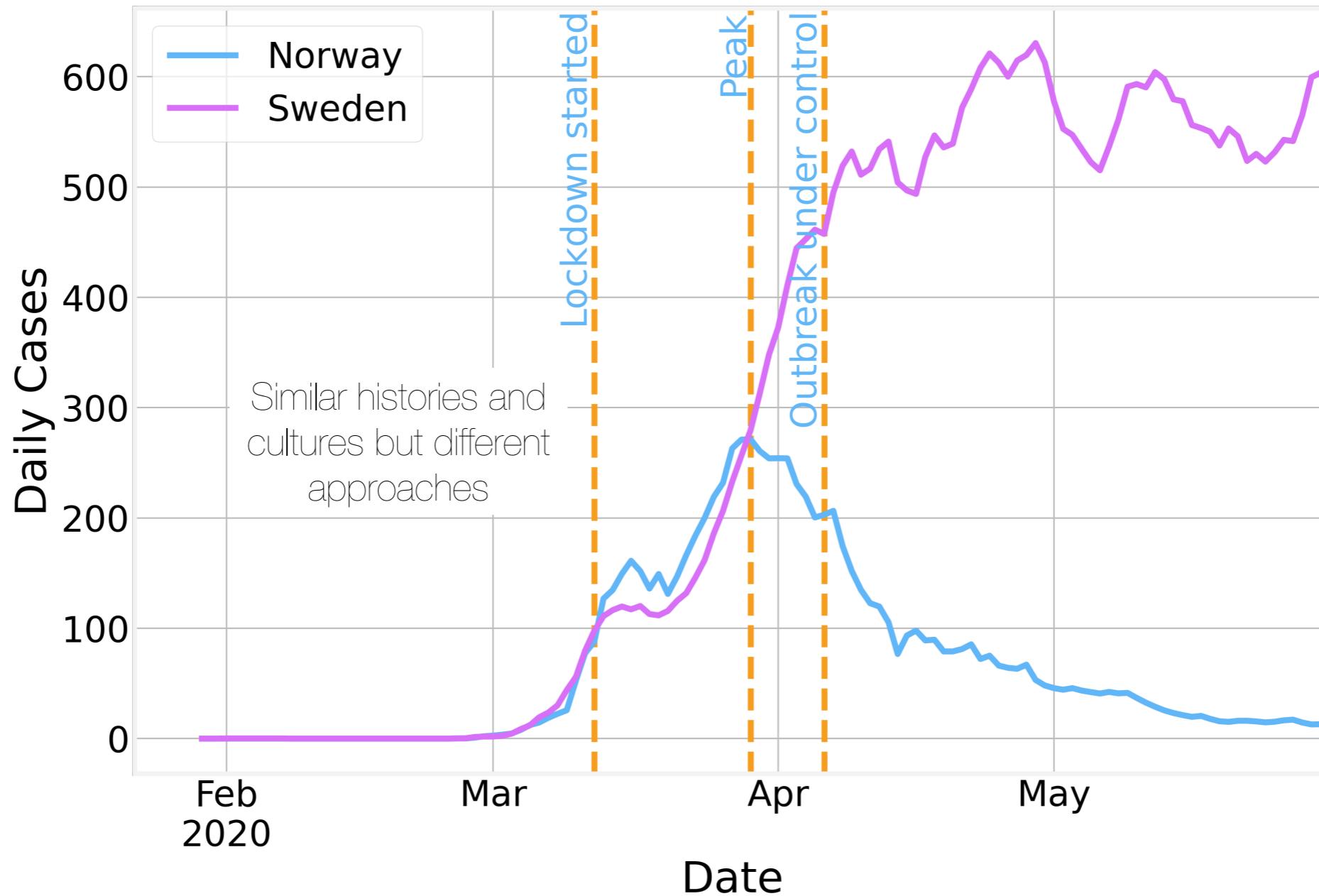
United States



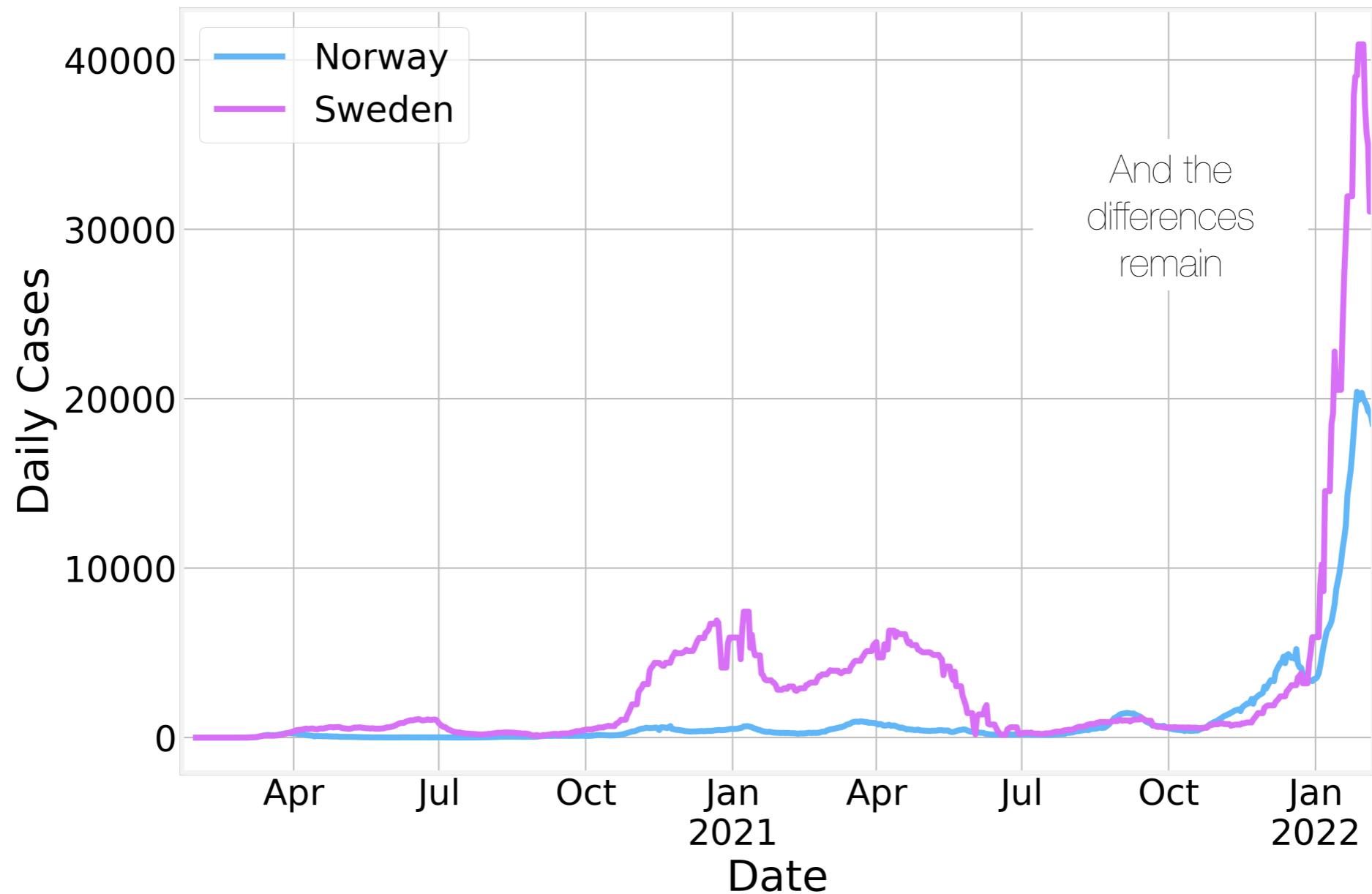
Cases by State



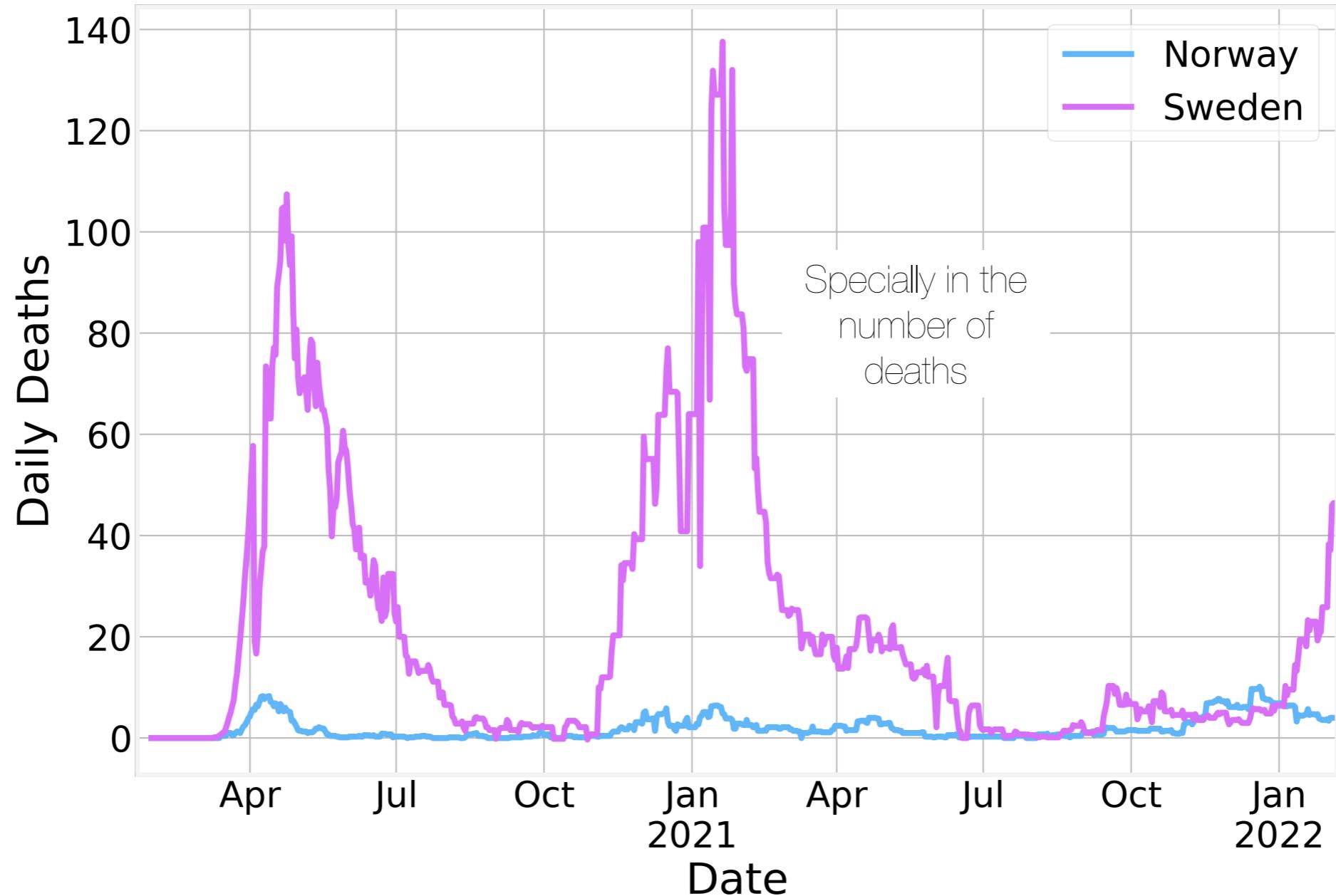
Country Comparisons



Country Comparisons



Country Comparisons

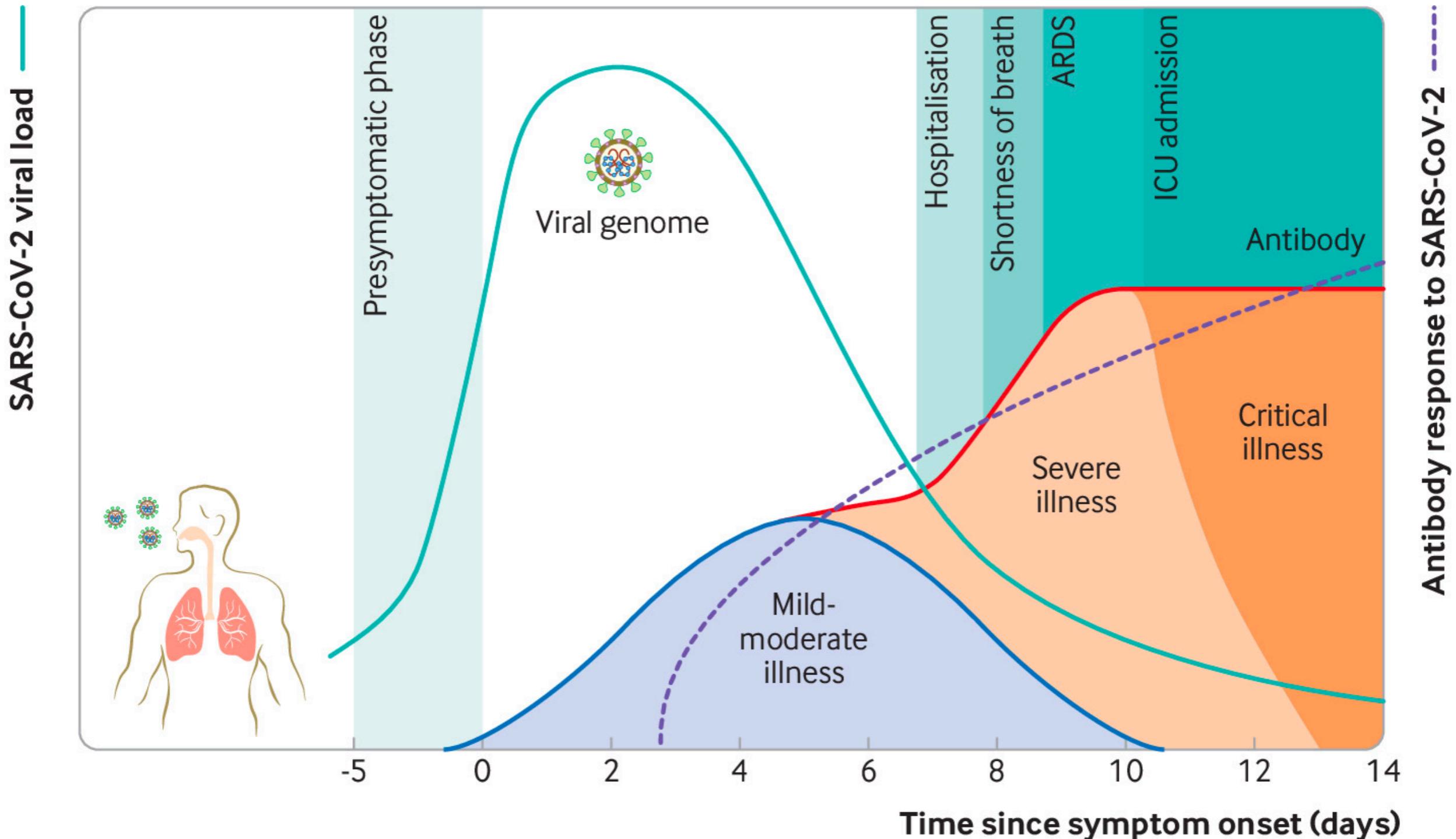




Code - Country Data Exploration
<https://github.com/DataForScience/CoVID19>

Viral Dynamics

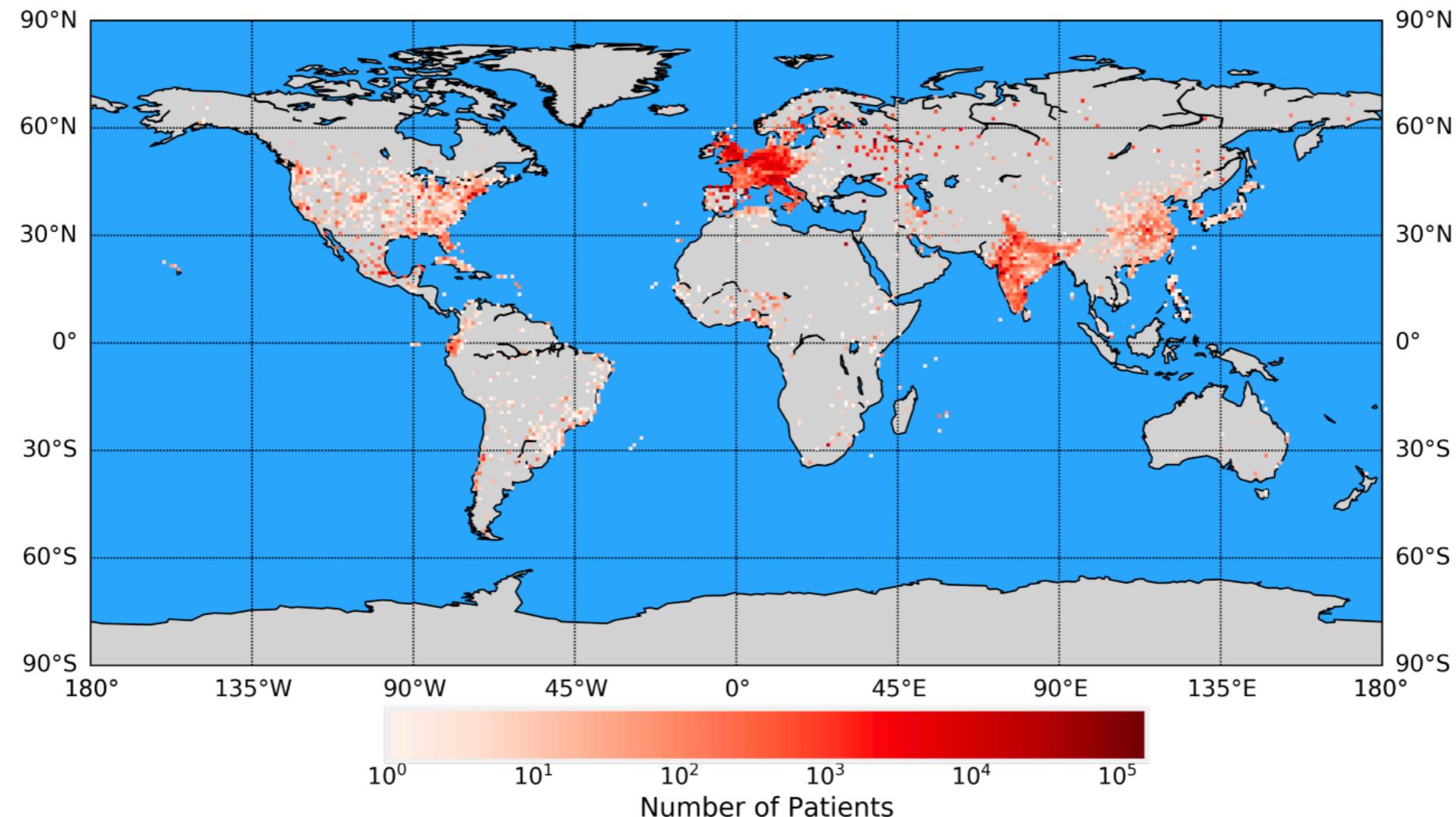
BMJ 371, m3862 (2020)



Individual Patient Data

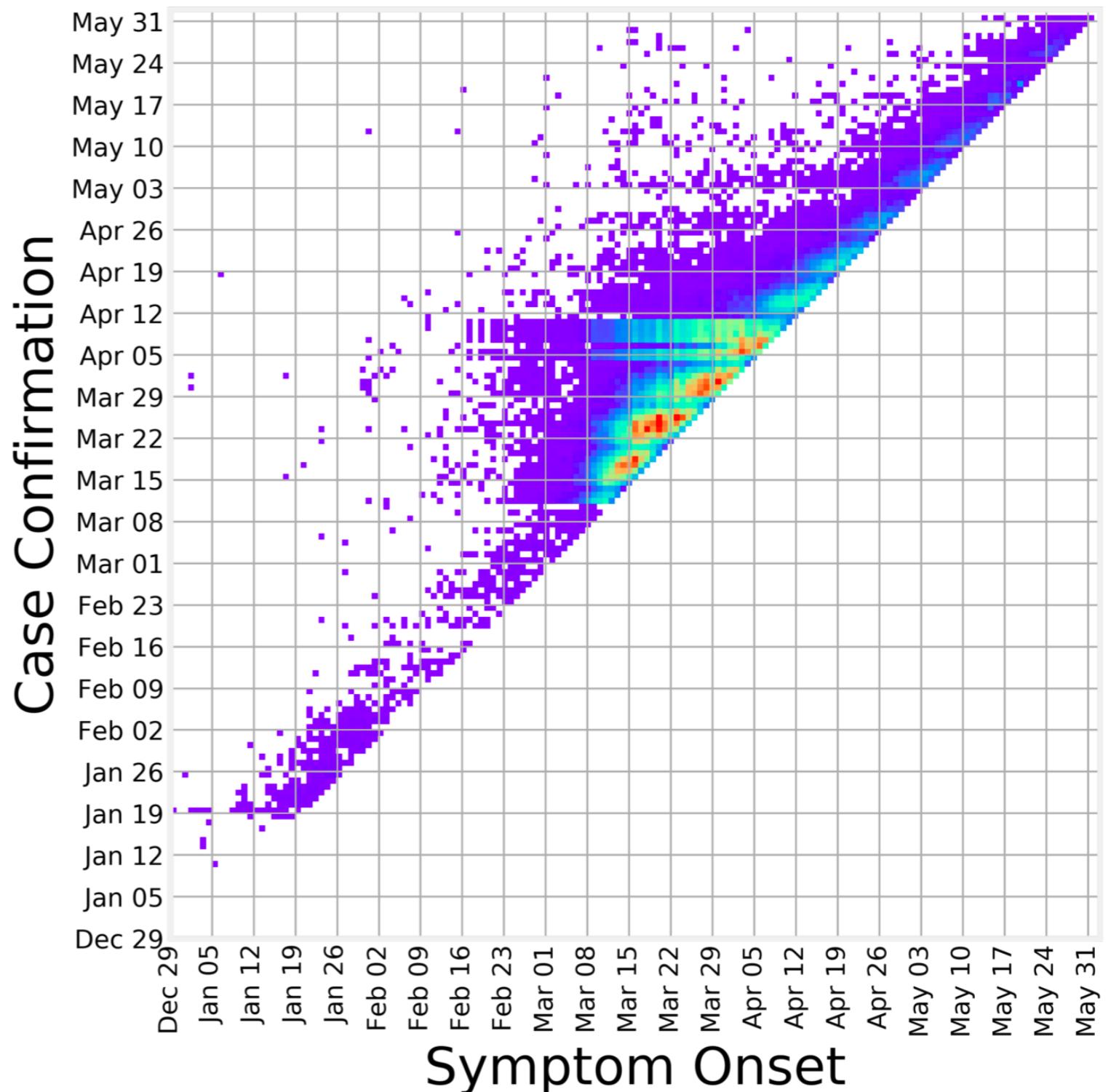
Scientific Data 7, 106 (2020)

- Detailed **line data** for hundreds of thousands of patients in various parts of the world
- Available for download here: <https://github.com/beoutbreakprepared/nCoV2019>
- Includes date of **symptom onset** and of **case confirmation**



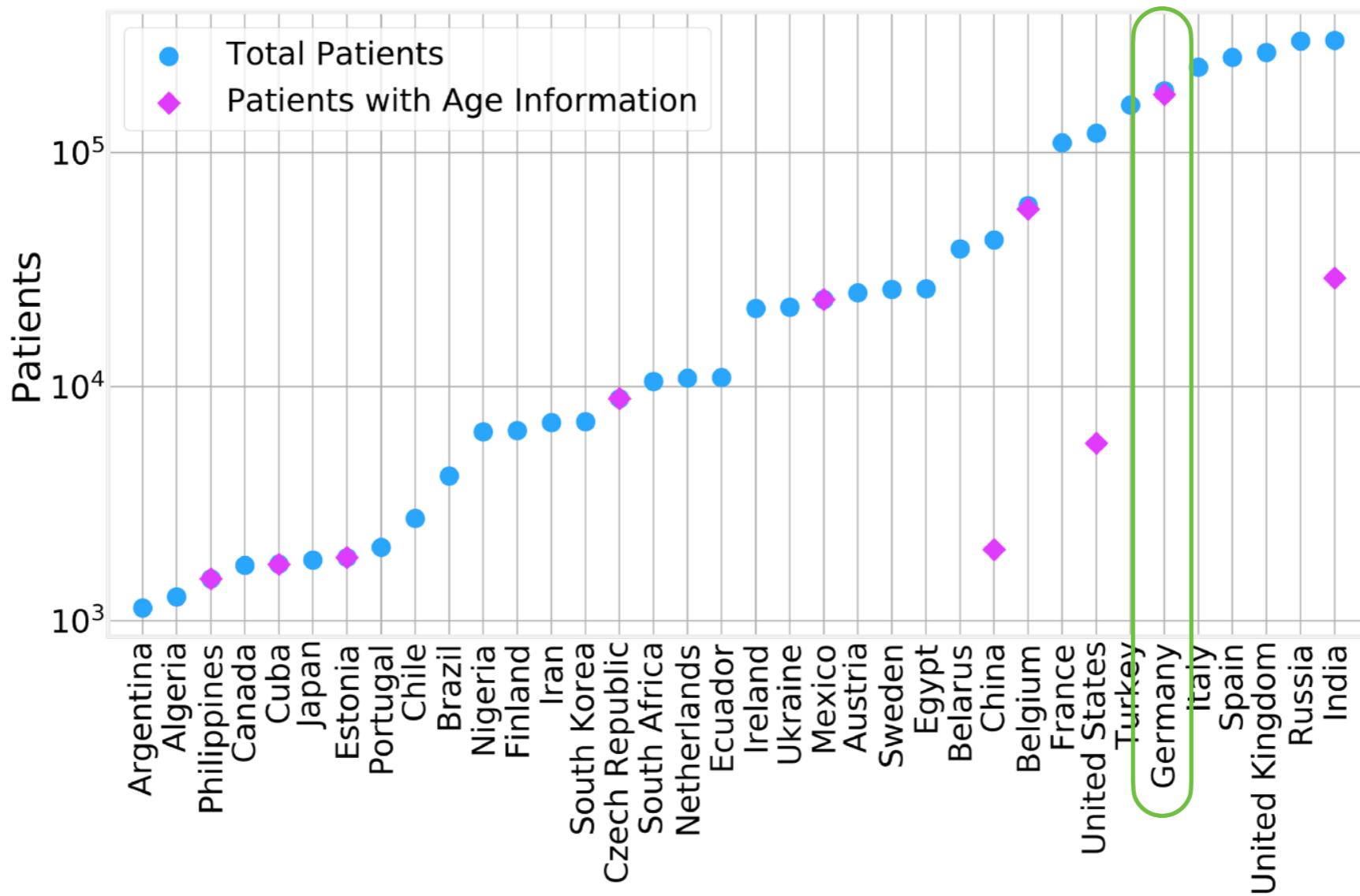
Case Confirmation Delay

- Symptom onset provides us with a better estimate of when infection might have occurred
- Variations in the lag can be due to testing policies, test availability, differences in severity, access to health care, etc



Individual Patient Data

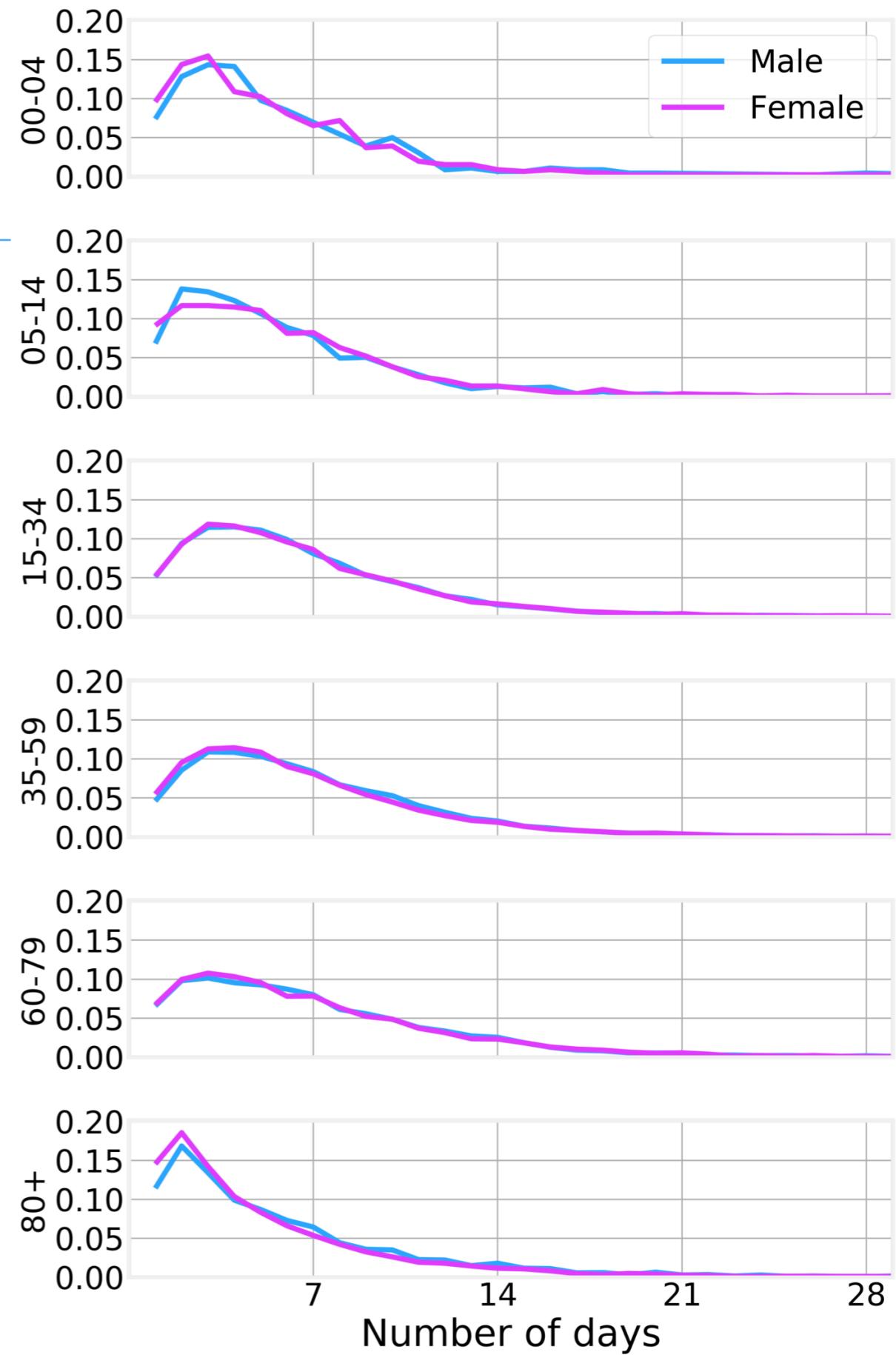
- Several countries provide age information as well
- Number of data points and data quality varies strongly from country to country



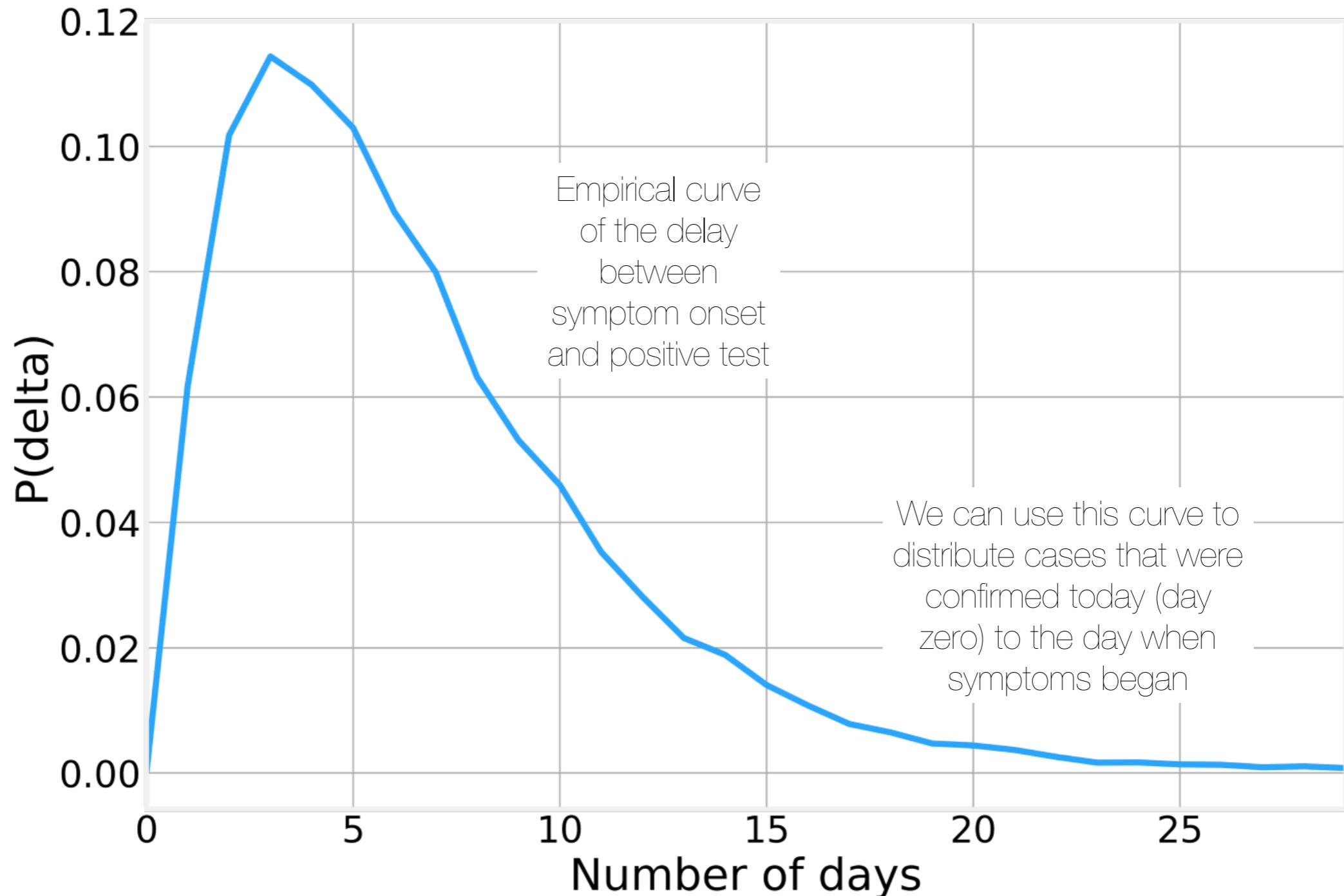
Germany Data

- Germany has the largest amount of gender and age group data
- No strong variation across ages or genders
- 3-4 days difference between Symptom onset and a positive test
- We compute the average across all data points

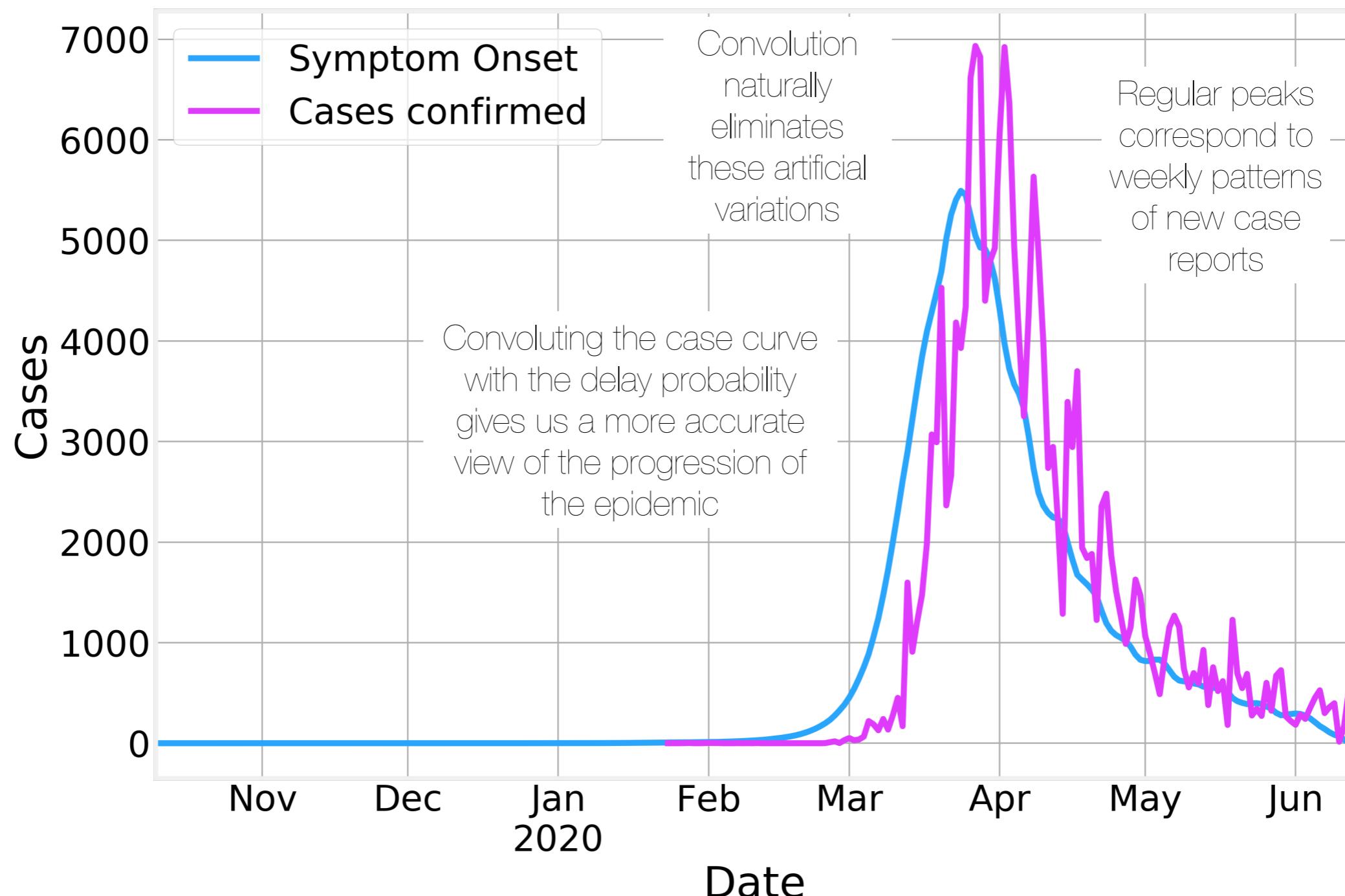
sex	female	male
age_group		
00-04	459	460
05-14	1095	1070
15-34	14911	13364
35-59	25832	23168
60-79	9818	11037
80+	5633	3913



Case Confirmation Delay

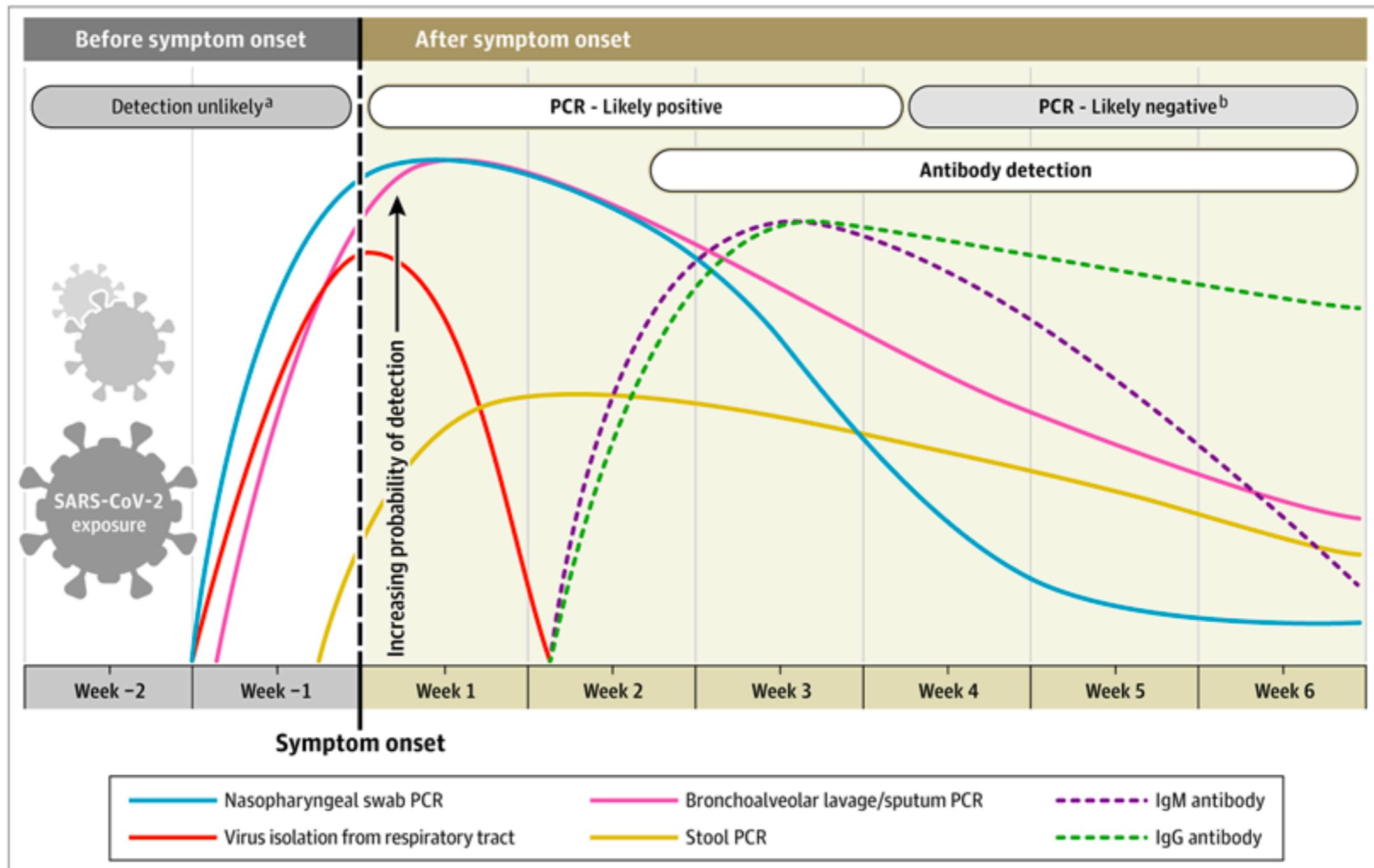


Symptom Onset Timeline



Testing

JAMA 323, 2249 (2020)



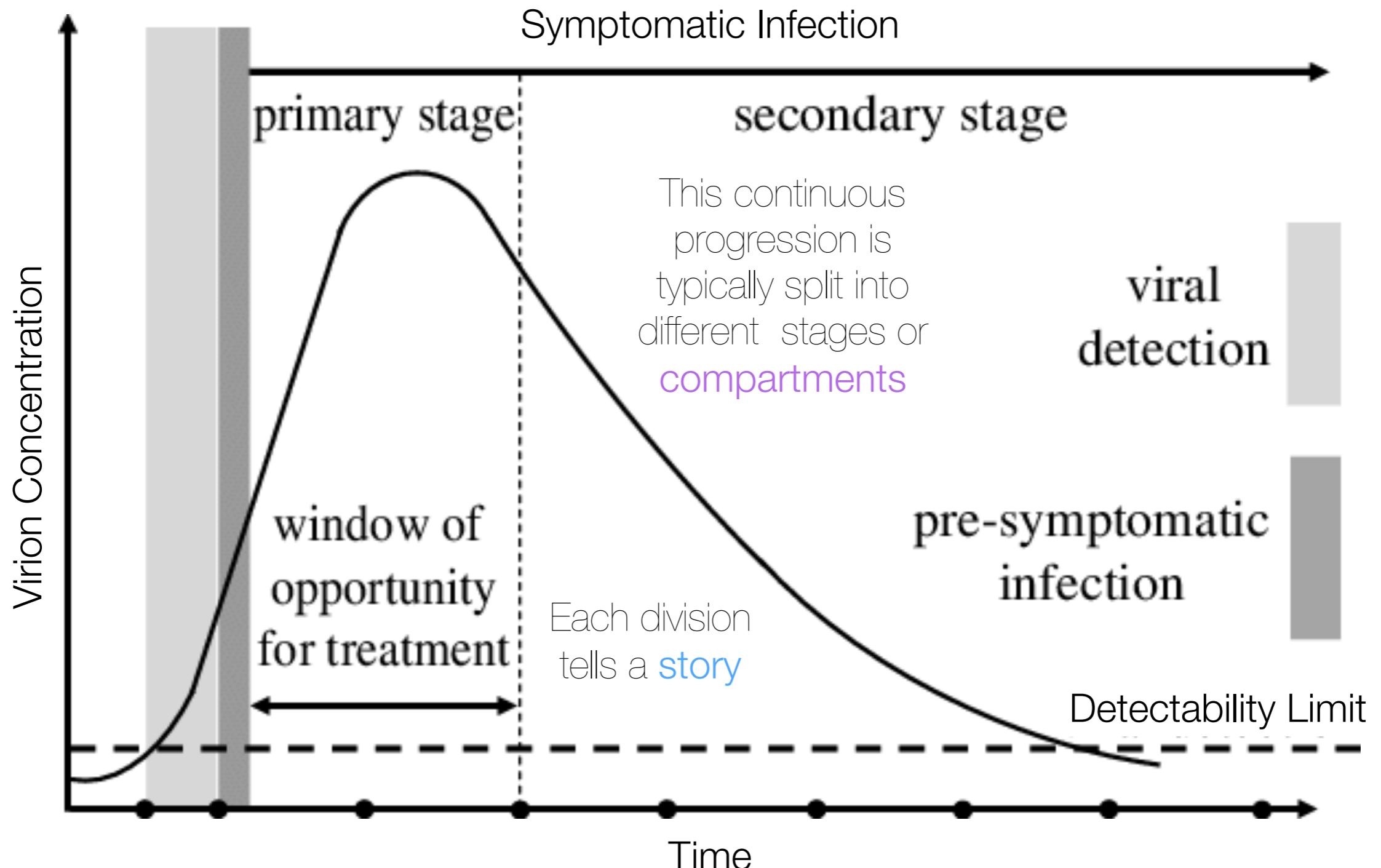


Code - Individual Data
<https://github.com/DataForScience/CoVID19>



2. Compartmental Models

Viral Dynamics (Cartoon)



Epidemic Stages (Population Level)

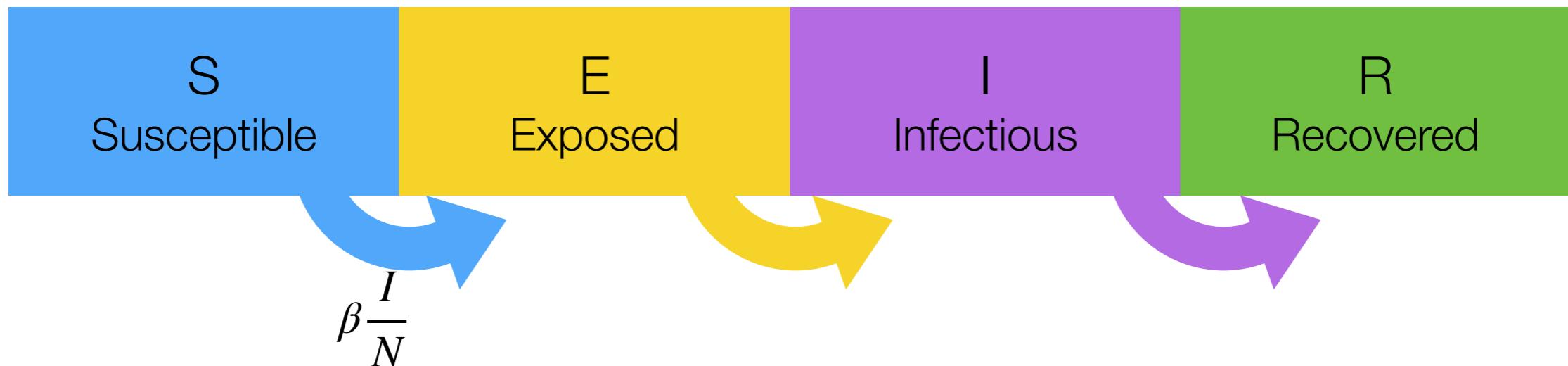
PNAS 106, 21484 (2009)



When a susceptible individual is exposed they incubate the disease for some time before becoming infectious and eventually recovering

Epidemic Stages (Population Level)

PNAS 106, 21484 (2009)



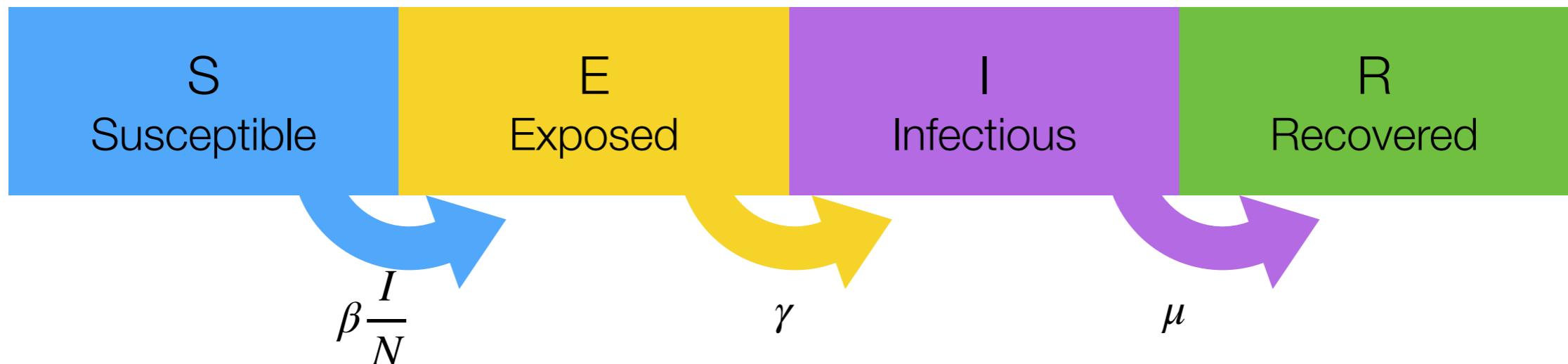
The more infectious individuals there are, the **more likely** someone is to get infected.

Each individual interacts with **every other**.

Each interactions has an (average) **fixed rate** of infection, β

Epidemic Stages (Population Level)

PNAS 106, 21484 (2009)



The more infectious individuals there are, the **more likely** someone is to get infected.

Each individual interacts with **every other**.

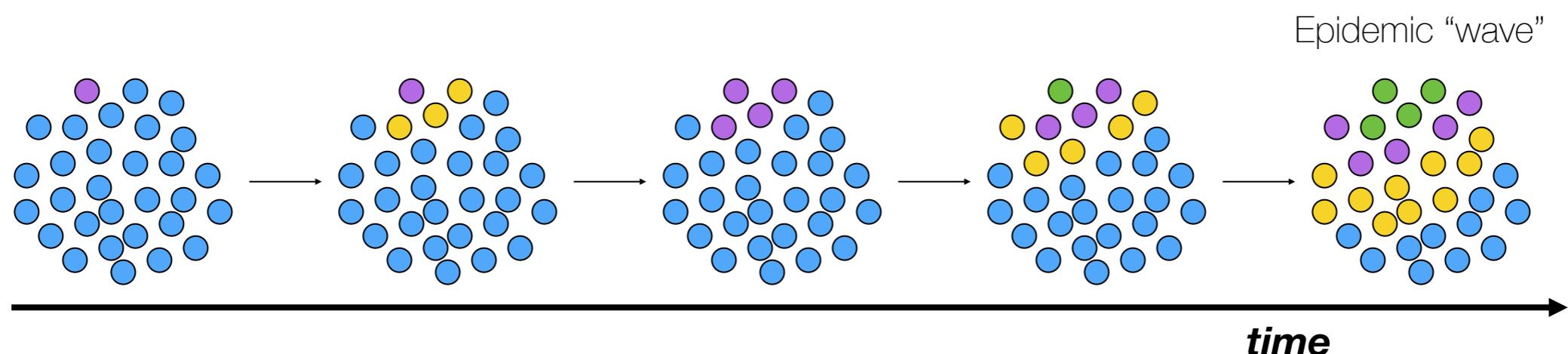
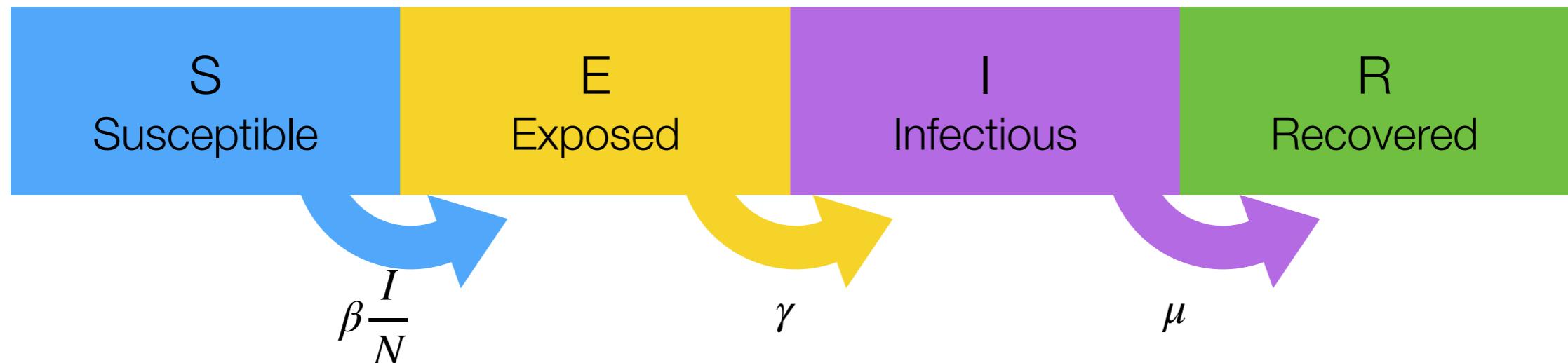
Each interactions has an (average) **fixed rate of infection**, β

Transitions from exposed to infectious occur at an (average) fixed rate, γ

Recovery occurs at an (average) fixed rate, μ

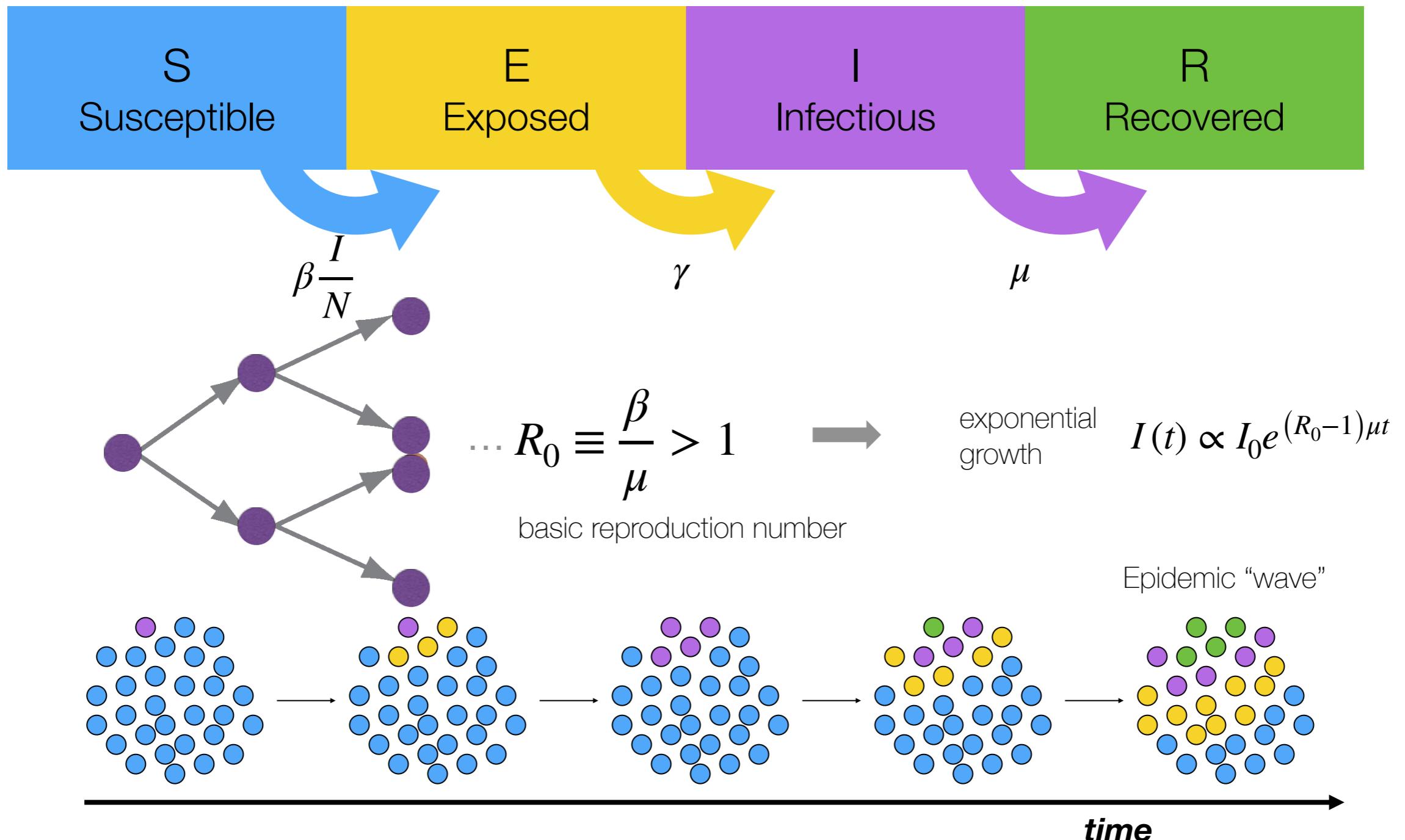
Epidemic Waves

PNAS 106, 21484 (2009)

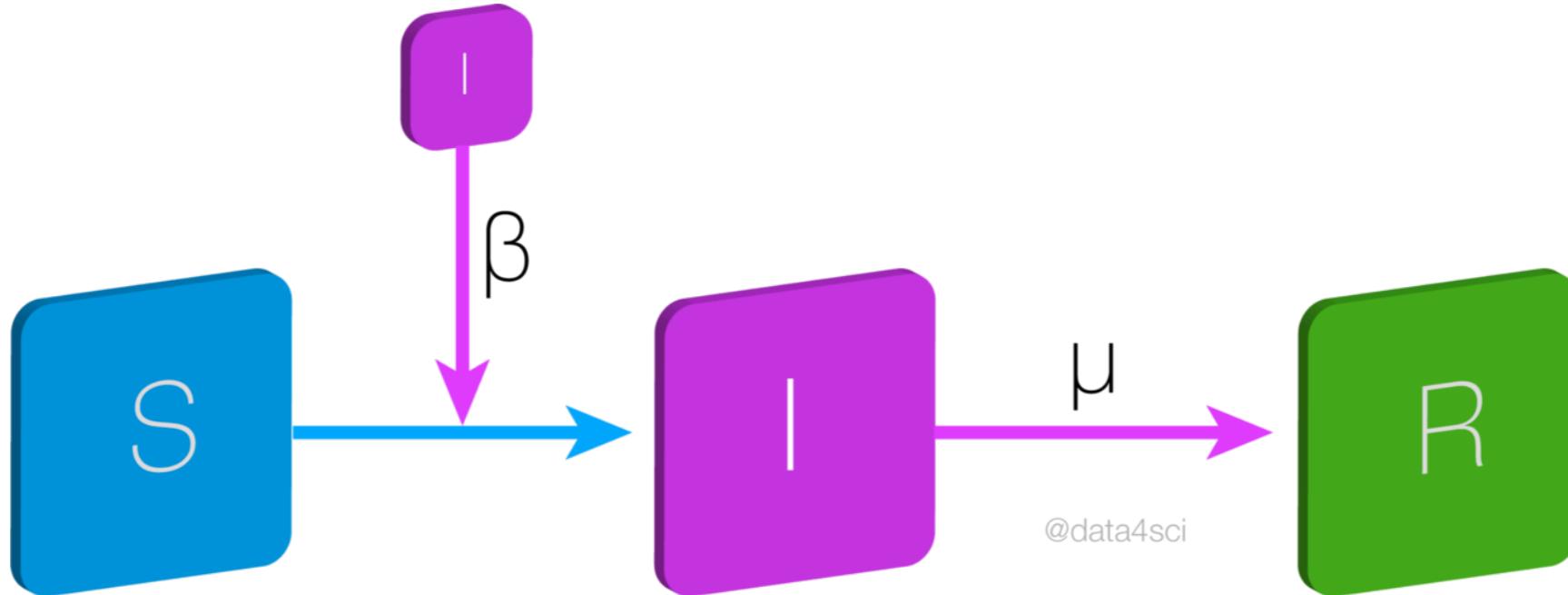


Epidemic Waves

PNAS 106, 21484 (2009)

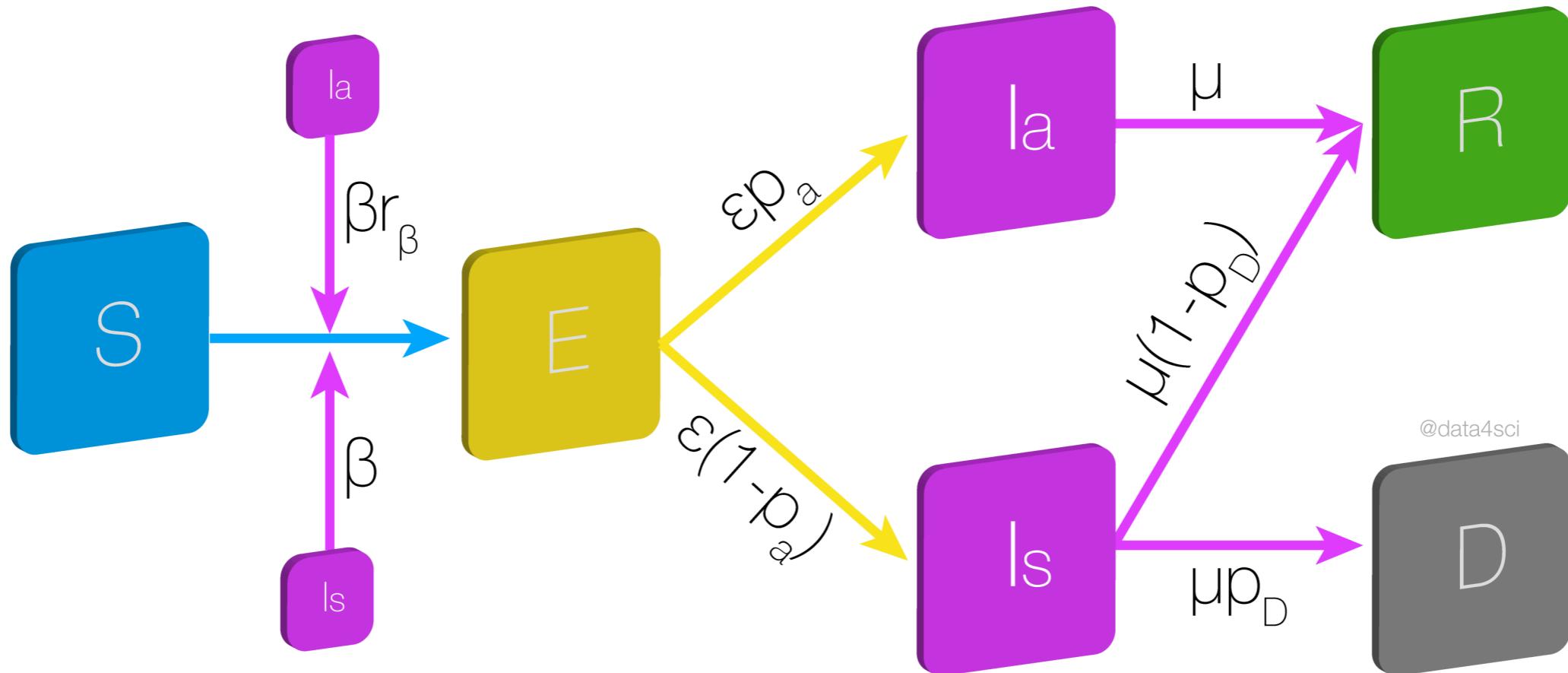


SIR Model



- One of the simplest models
- You immediately become infectious (no incubation period)
- Every one eventually recovers

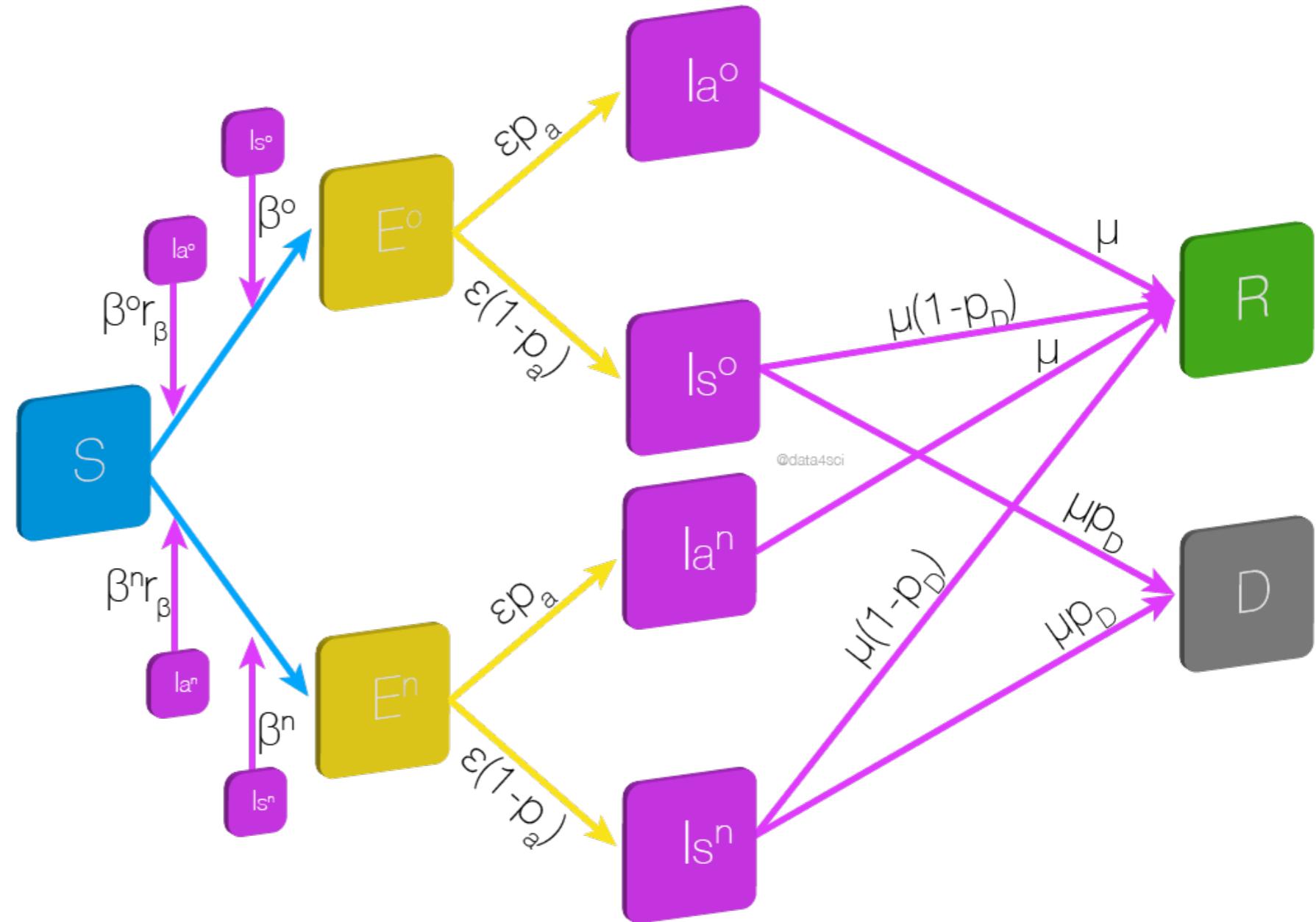
Simple CoVID-19 Model



- Includes incubation period (Exposed compartment)
- Distinguishes between Symptomatic and Asymptomatic cases
- Includes the possibility of Death instead of Recovery

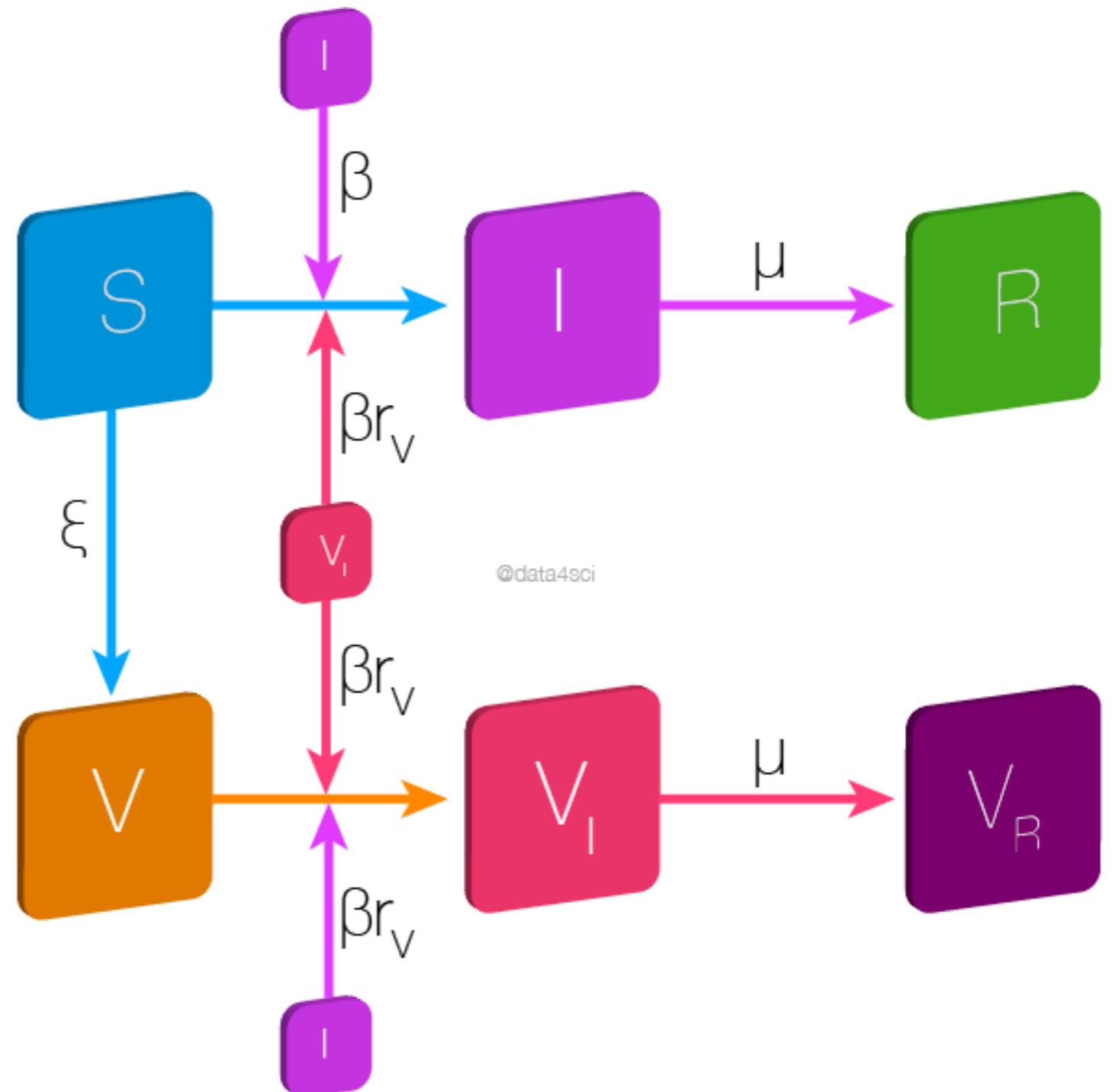
CoVID-19 Model with Competing Strains

- Both strains have symptomatic and asymptomatic cases and an incubation period
- Only difference is in infectiousness, $\beta^{(n,o)}$



Simplest CoVID-19 Model with Vaccination

- No incubation period
- No Asymptomatic cases
- Vaccinated individuals can still be infected and become infectious, with a reduction in rate of r_v
- Only healthy people who never caught the virus can be vaccinated at a rate of ξ

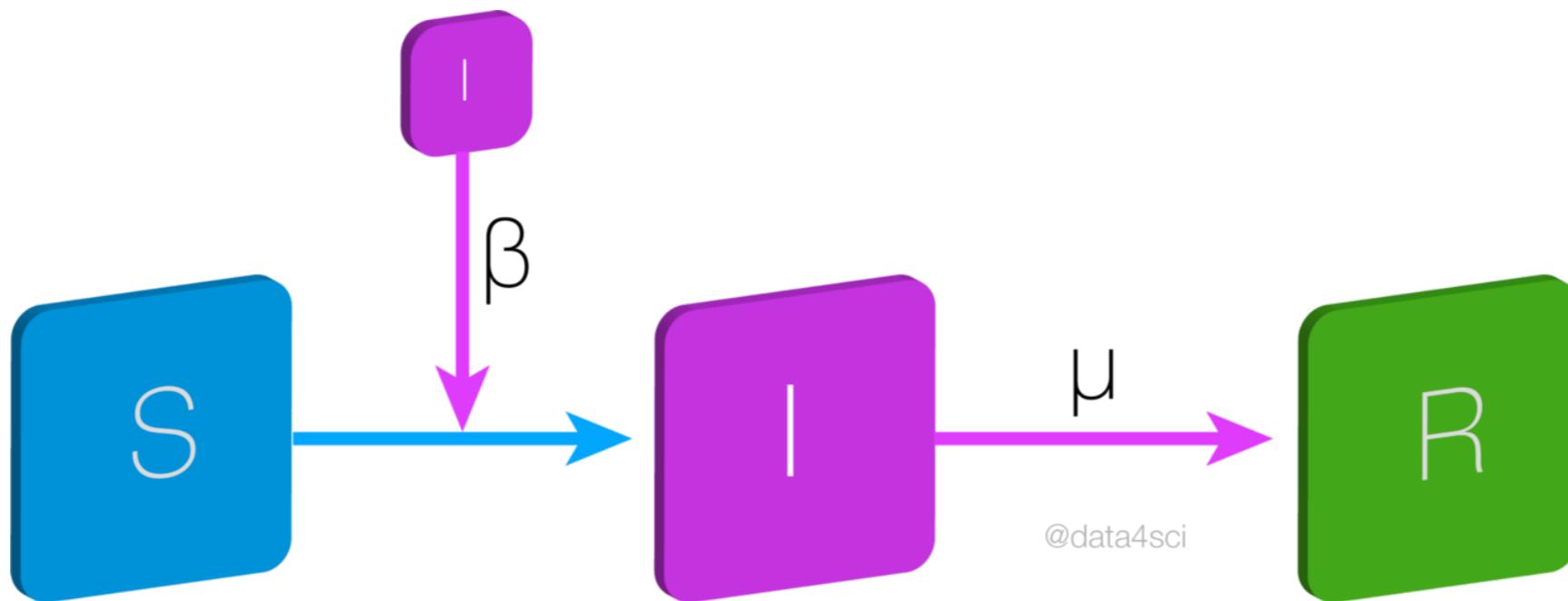


@data4sci

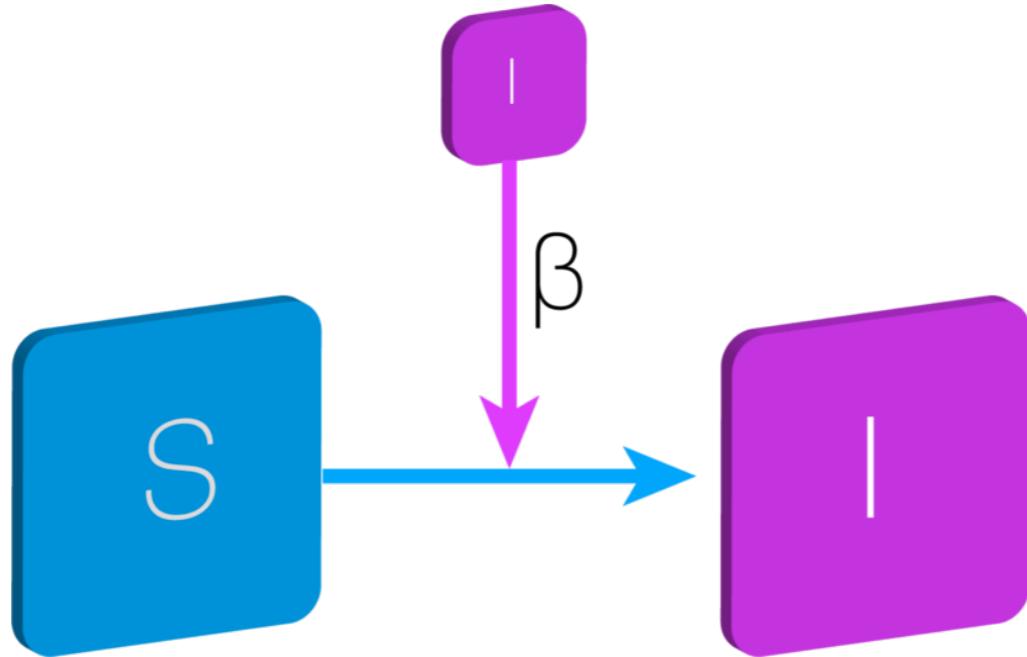
Model Structure

- Compartmental models can be represented as a **Directed Acyclical Graph** (DAG)
- Each compartment is a **node** and each **edge** is a transition between two compartments
- Two types of transitions between states represented by two kinds of edges:
 - **Spontaneous** - depends only on the population of the current compartment γE
 - **Interaction** - depends on the population of a third compartment $\beta S \frac{I}{N}$
- Transitions represent the changes in compartment populations
- Each edge can be trivially mapped to a mathematical expression

Model Structure



Model Structure



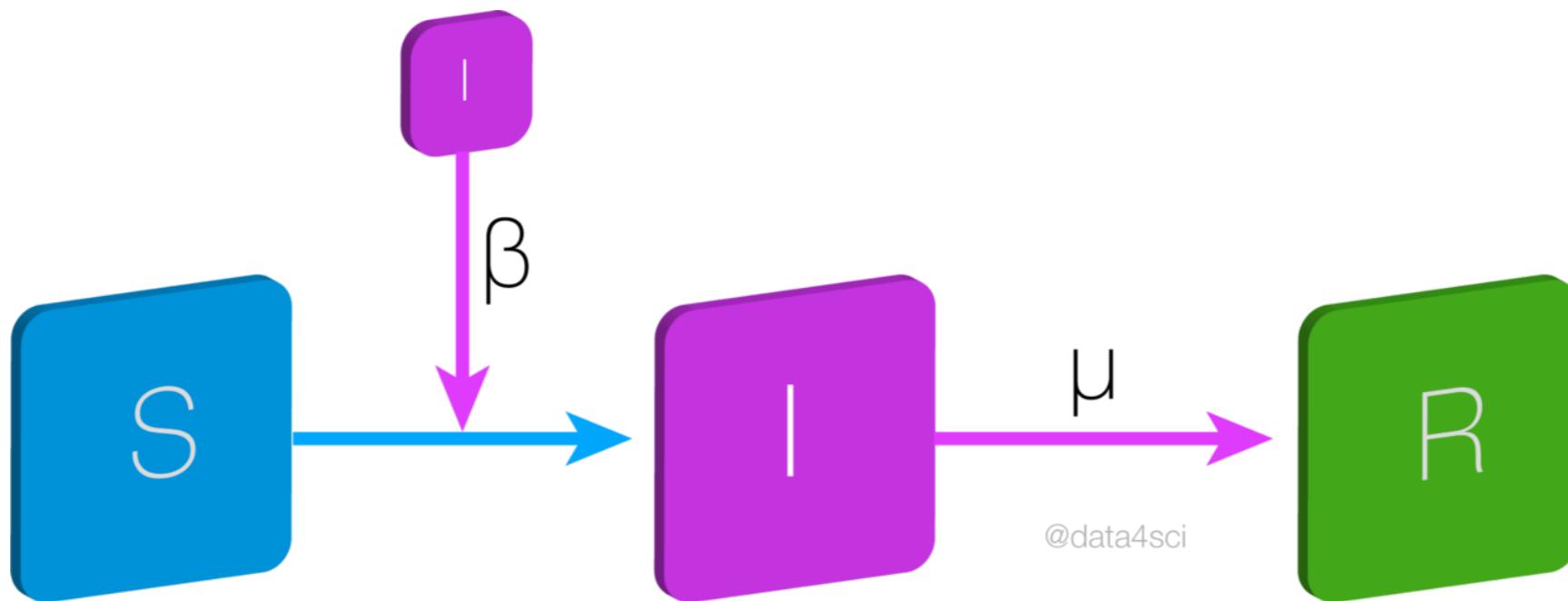
Each individual removed from S is added to I

$$\frac{dS}{dt} = -\beta S \frac{I}{N}$$

The more infectious individuals the higher the rate of infection

$$\frac{dI}{dt} = +\beta S \frac{I}{N}$$

Model Structure



$$\frac{dS}{dt} = -\beta S \frac{I}{N}$$

$$\frac{dI}{dt} = +\beta S \frac{I}{N} - \mu I$$

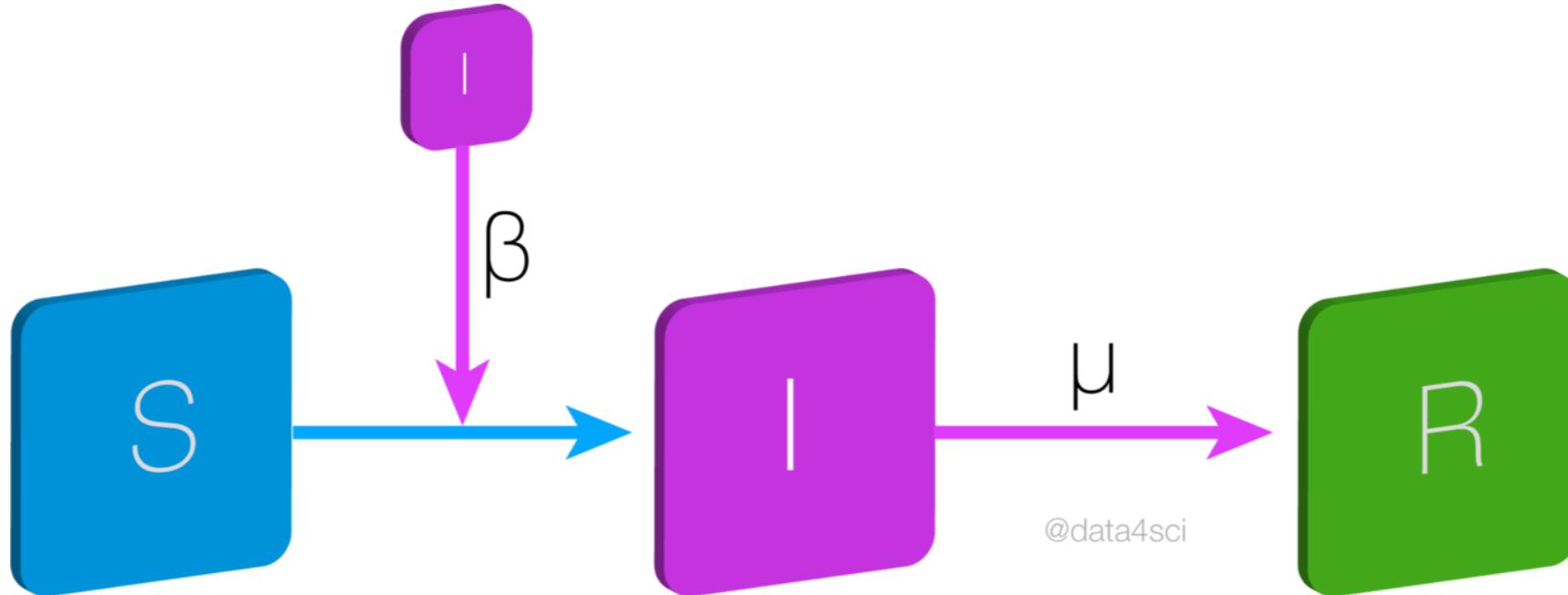
$$\frac{dR}{dt} = +\mu I$$

$$\frac{dS}{dt} + \frac{dI}{dt} + \frac{dR}{dt} = 0$$

Recovered
individuals move
from I to R

Total population
is constant

Model Structure



$$\frac{dS}{dt} = -\beta S \frac{I}{N}$$

$$\frac{dI}{dt} = +\beta S \frac{I}{N} - \mu I$$

$$\frac{dR}{dt} = +\mu I$$

$$\frac{dS}{dt} + \frac{dI}{dt} + \frac{dR}{dt} = 0$$

Recovered
individuals move
from I to R

Total population
is constant

In the beginning:

$$S \sim N$$
$$I = I_0 \sim 1$$

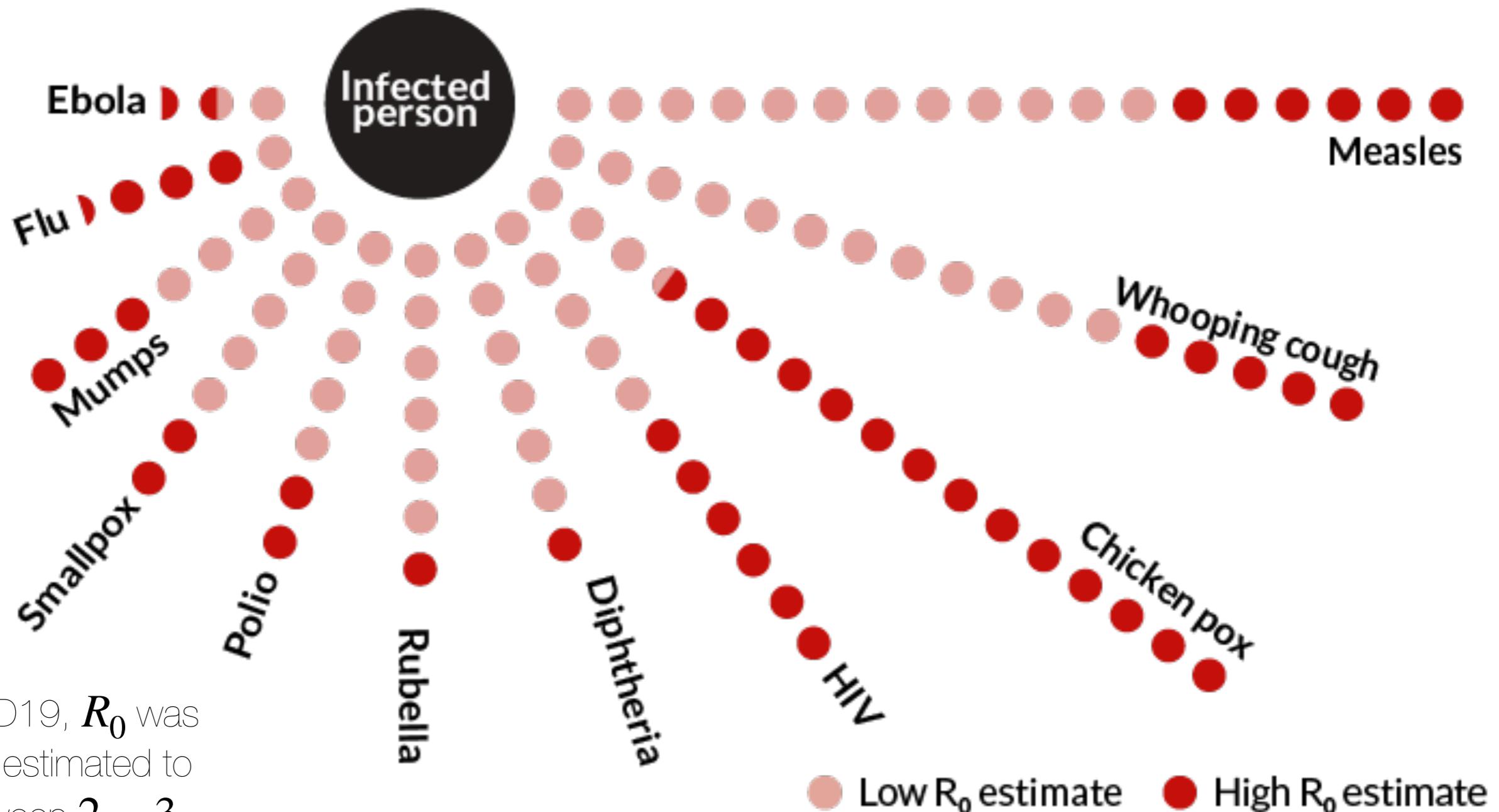
Therefore:

Basic
Reproductive
Number

$$\frac{dI}{dt} \sim (\beta - \mu) \frac{I}{N} \rightarrow I(t) \propto I_0 e^{\mu(\frac{\beta}{\mu} - 1)t}$$

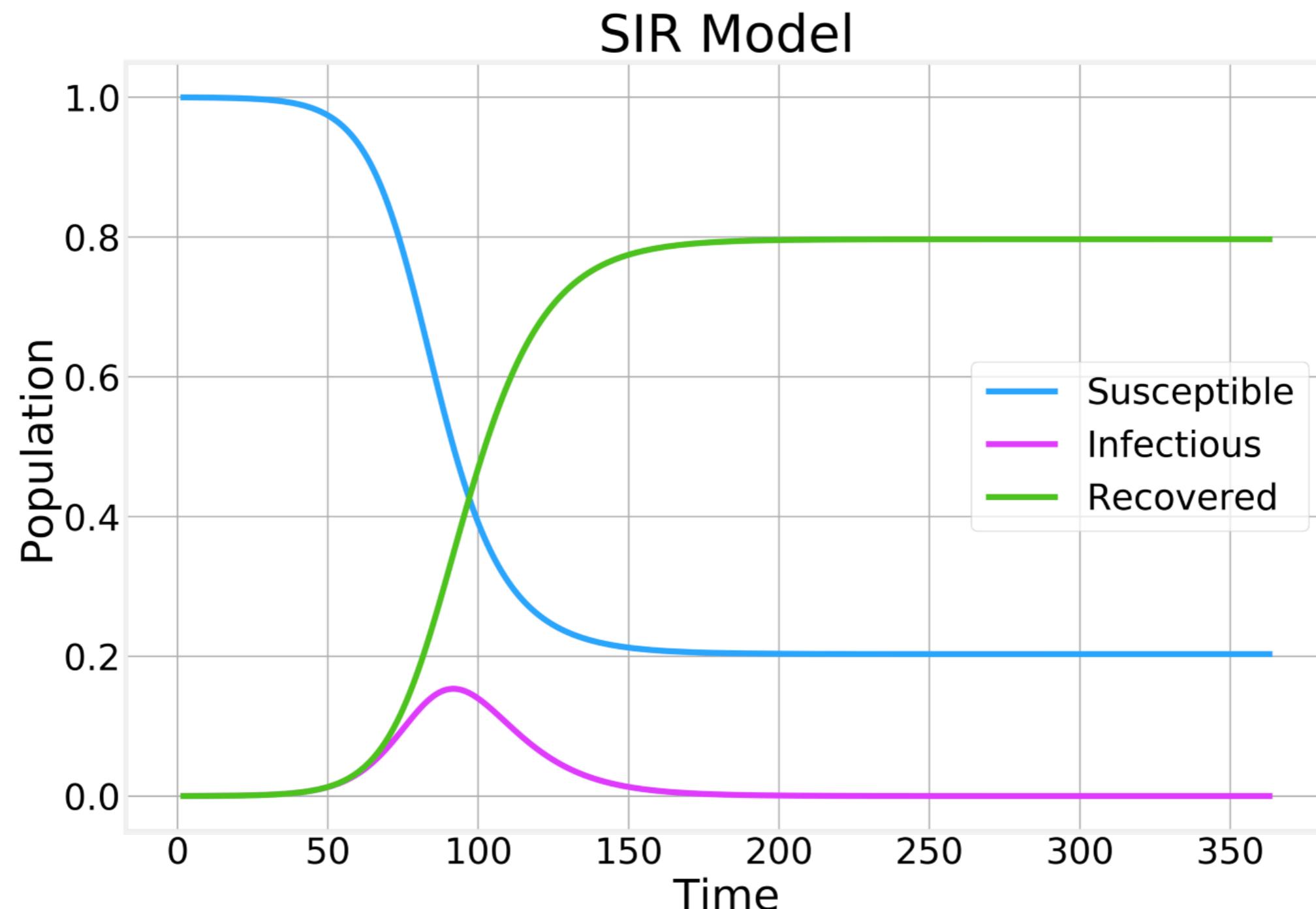
R_0 - Basic Reproductive Number

<https://www.sciencenewsforstudents.org/article/numbers-how-infectious-measles-and-other-diseases-spread>

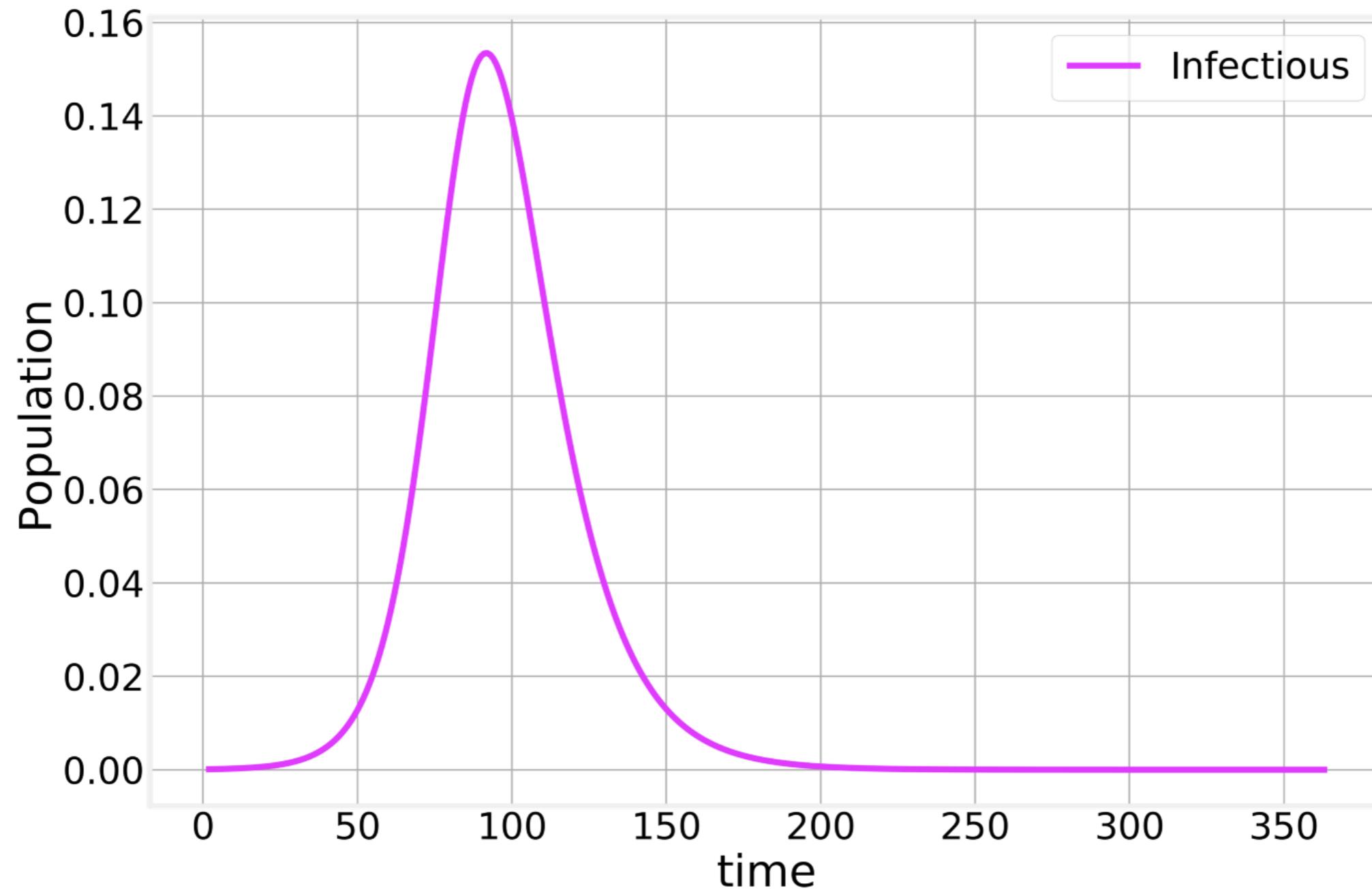


For COVID19, R_0 was originally estimated to be between 2 – 3

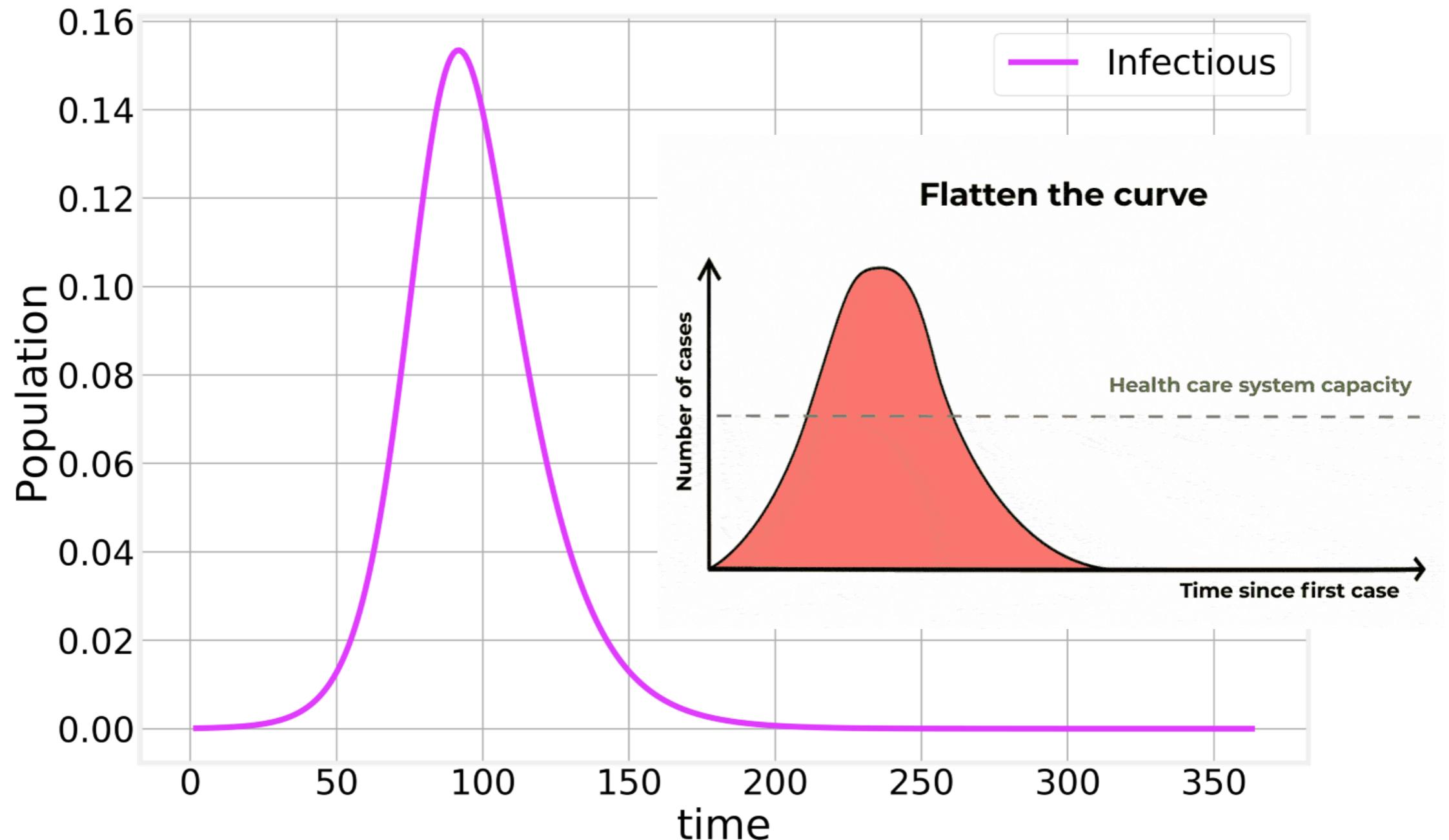
SIR Model



SIR Model



SIR Model



- Create a utility class to define the model DAG

```
class EpiModel(object):
    """Simple Epidemic Model Implementation

    Provides a way to implement and numerically integrate
    """

    def __init__(self, compartments=None):
        self.transitions = nx.MultiDiGraph()
        self.seasonality = None

        if compartments is not None:
            self.transitions.add_nodes_from([comp for comp in compartments])

    def add_interaction(self, source, target, agent, rate):
        self.transitions.add_edge(source, target, agent=agent, rate=rate)

    def add_spontaneous(self, source, target, rate):
        self.transitions.add_edge(source, target, rate=rate)
```

EpiModel.py

- We use SciPy `odeint` to numerically integrate the model
- `_new_cases()` is where we compute the right hand side of the equations above

```
def integrate(self, timesteps, t_min=1, seasonality=None, **kwargs):  
    """Numerically integrate the epidemic model"""  
    pos = {comp: i for i, comp in enumerate(self.transitions.nodes())}  
    population=np.zeros(len(pos))  
  
    for comp in kwargs:  
        population[pos[comp]] = kwargs[comp]  
  
    time = np.arange(t_min, t_min+timesteps, 1)  
  
    self.seasonality = seasonality  
    self.values_ = pd.DataFrame(odeint(self._new_cases,  
                                       population,  
                                       time,  
                                       args=(pos,),  
                                       columns=pos.keys(), index=time))
```

```
def _new_cases(self, population, time, pos):
    """Internal function used by integration routine"""
    diff = np.zeros(len(pos))
    N = np.sum(population)

    for edge in self.transitions.edges(data=True):
        source = edge[0]
        target = edge[1]
        trans = edge[2]

        rate = trans['rate']*population[pos[source]]

        if 'agent' in trans:
            agent = trans['agent']
            rate *= population[pos[agent]]/N

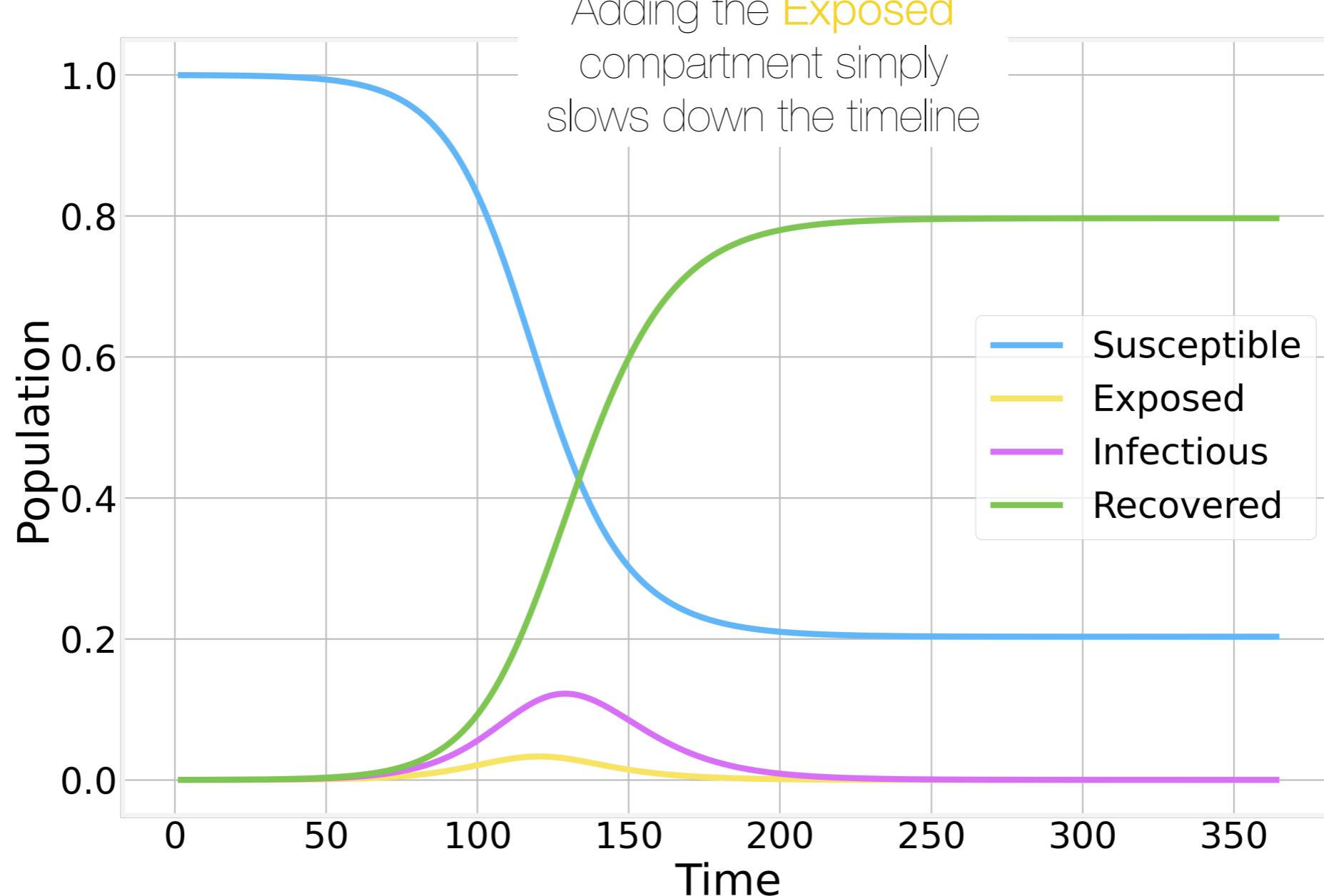
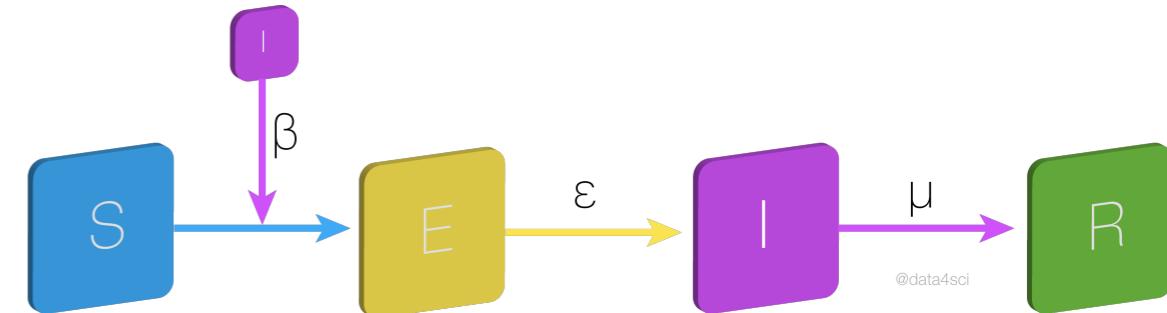
        diff[pos[source]] -= rate
        diff[pos[target]] += rate
```

Rate for spontaneous transactions

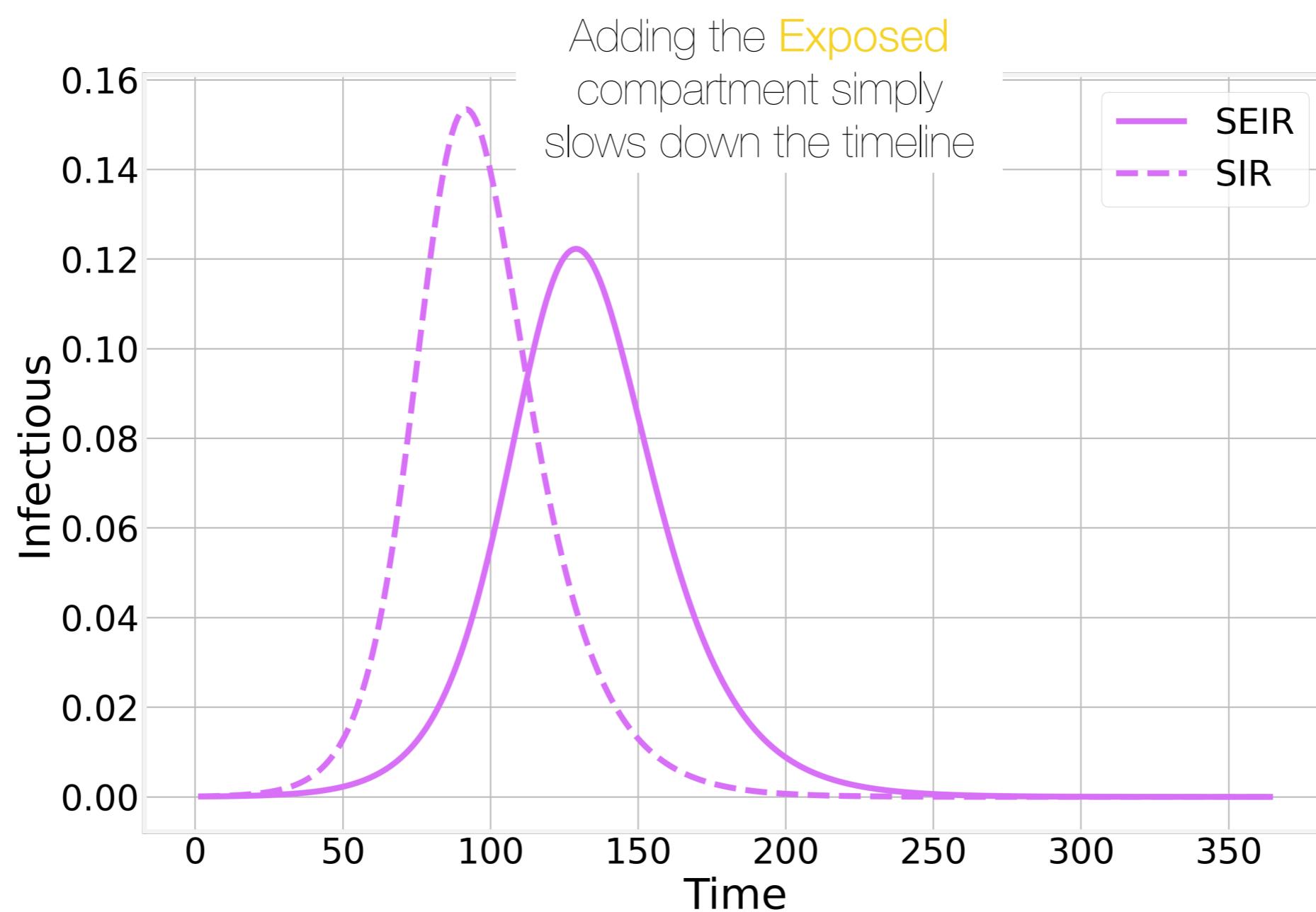
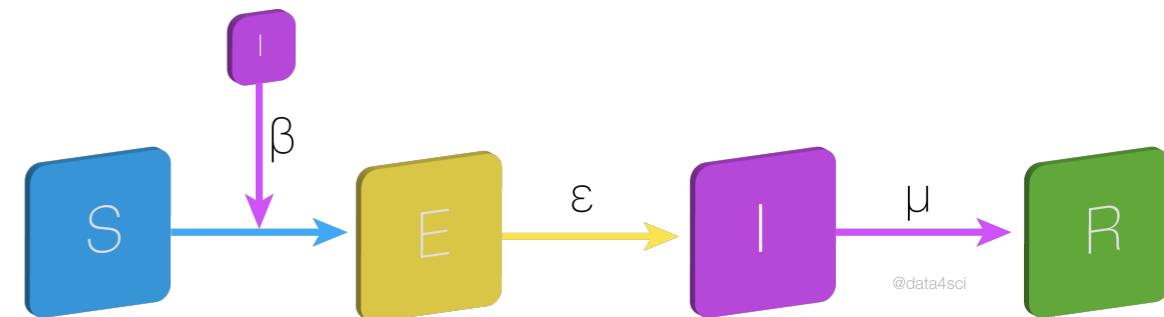
Modify for interactive transaction

Update compartment populations

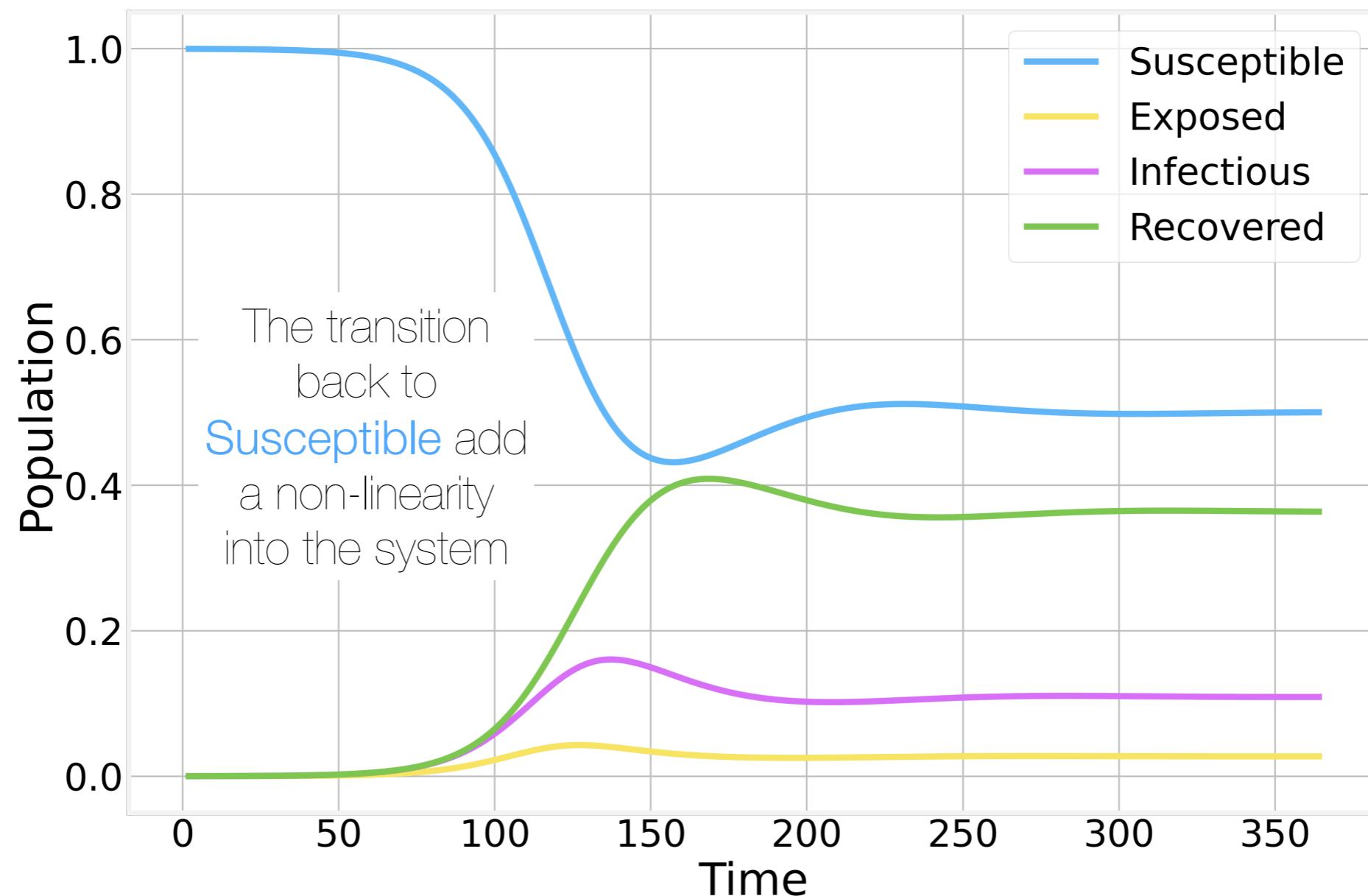
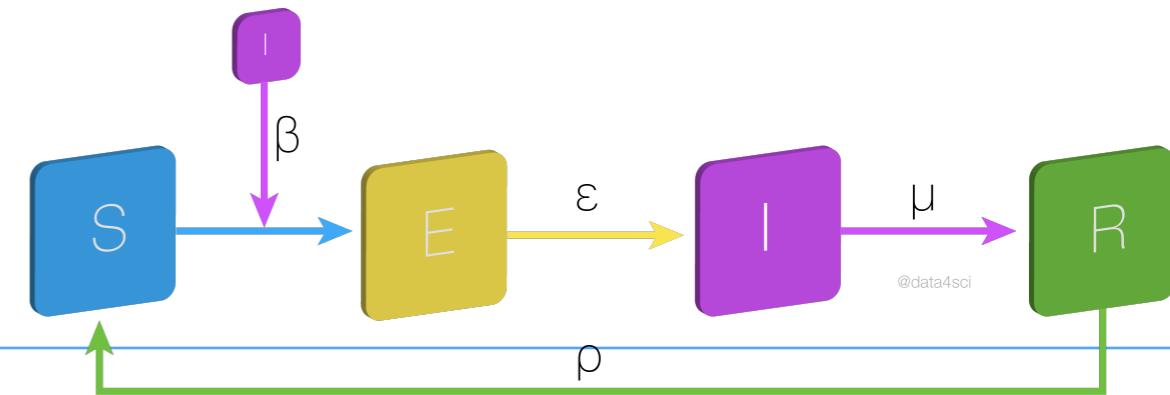
SIR - Incubation Period



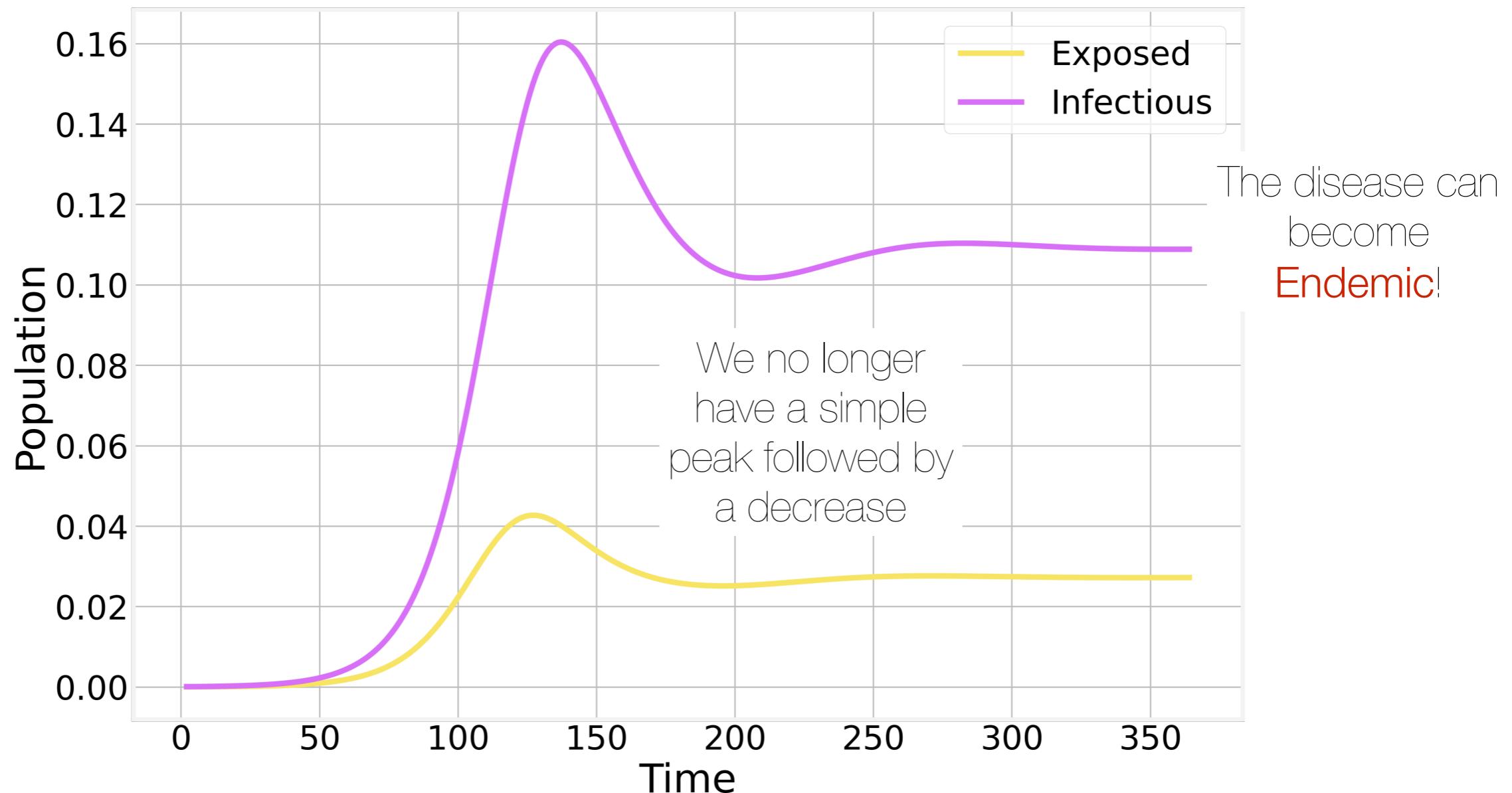
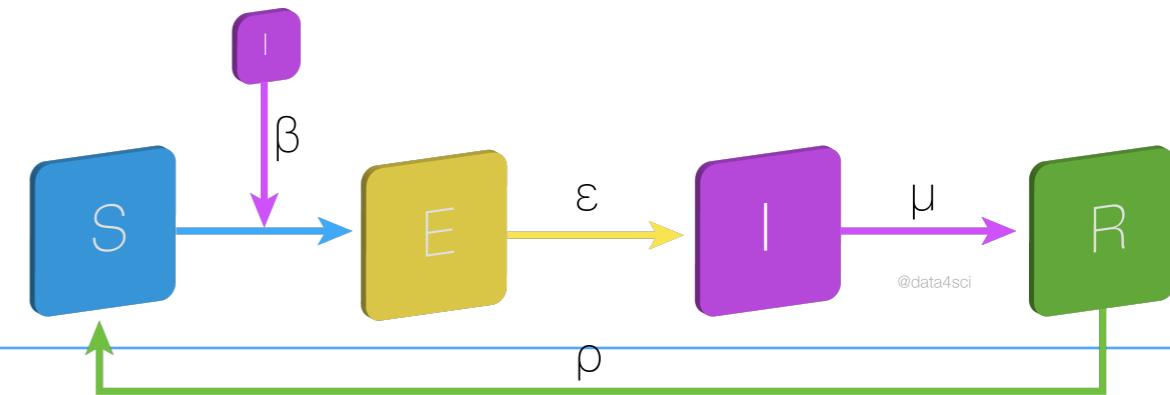
SIR - Incubation Period



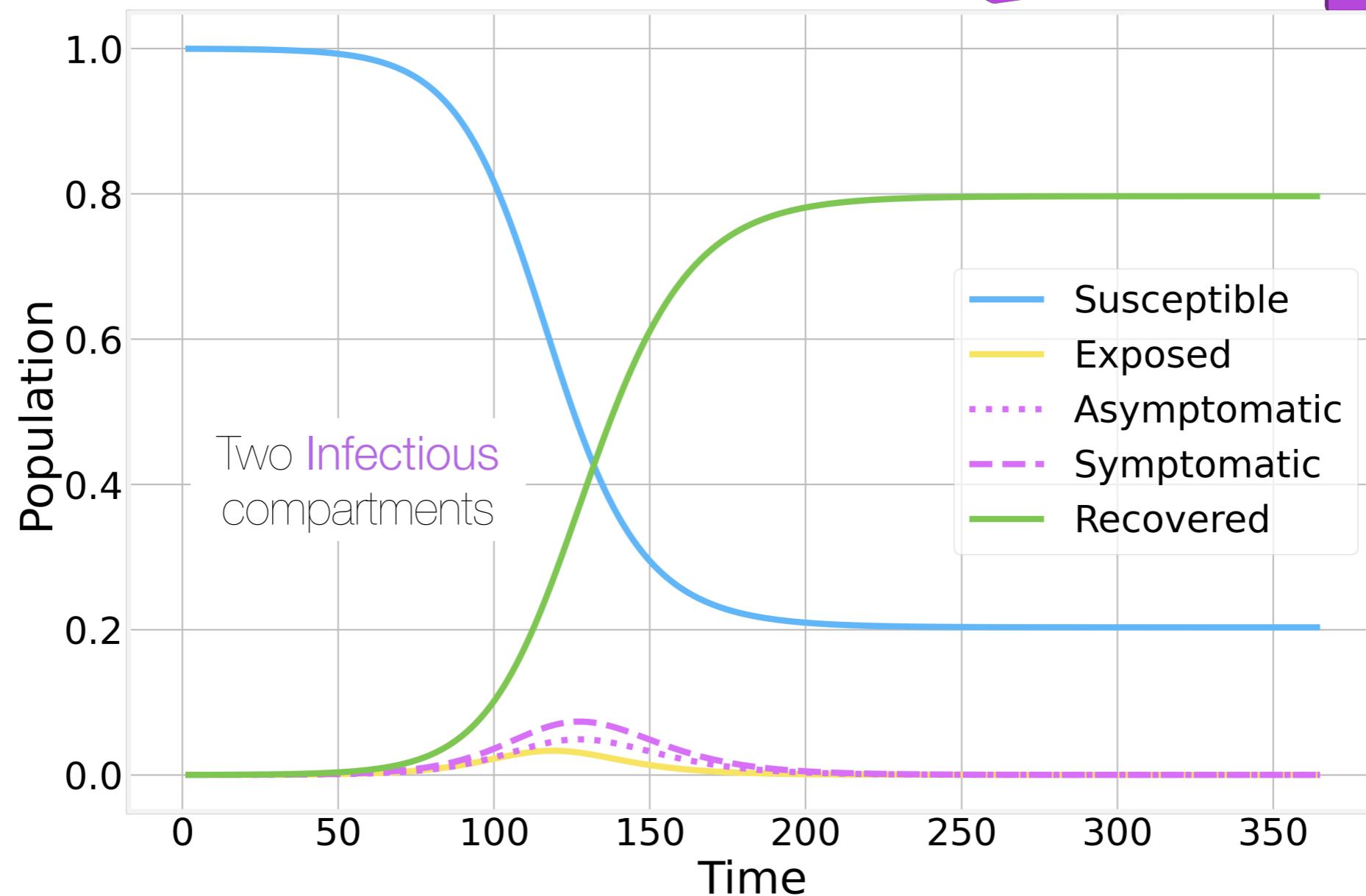
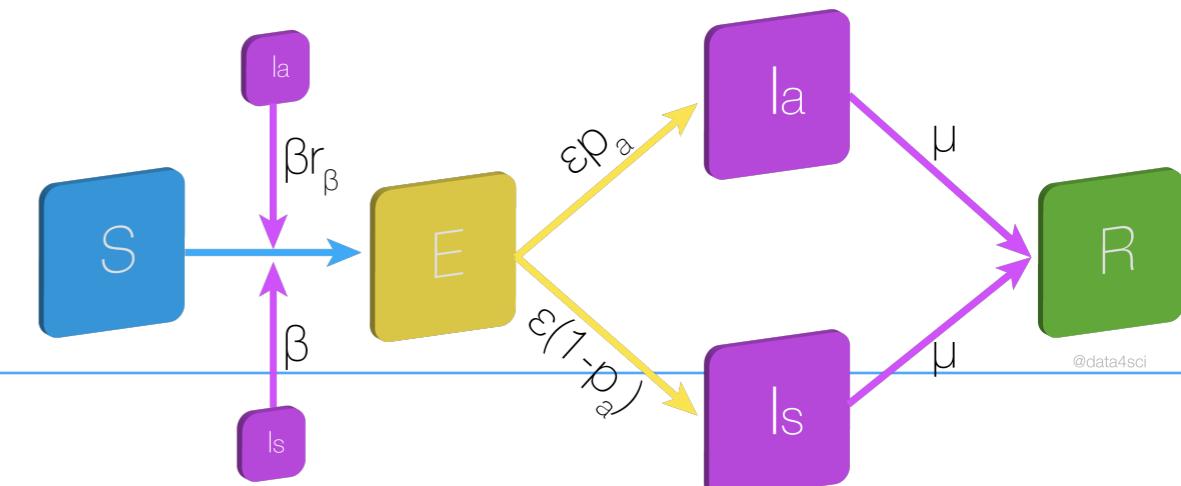
SEIRS - Waning Immunity



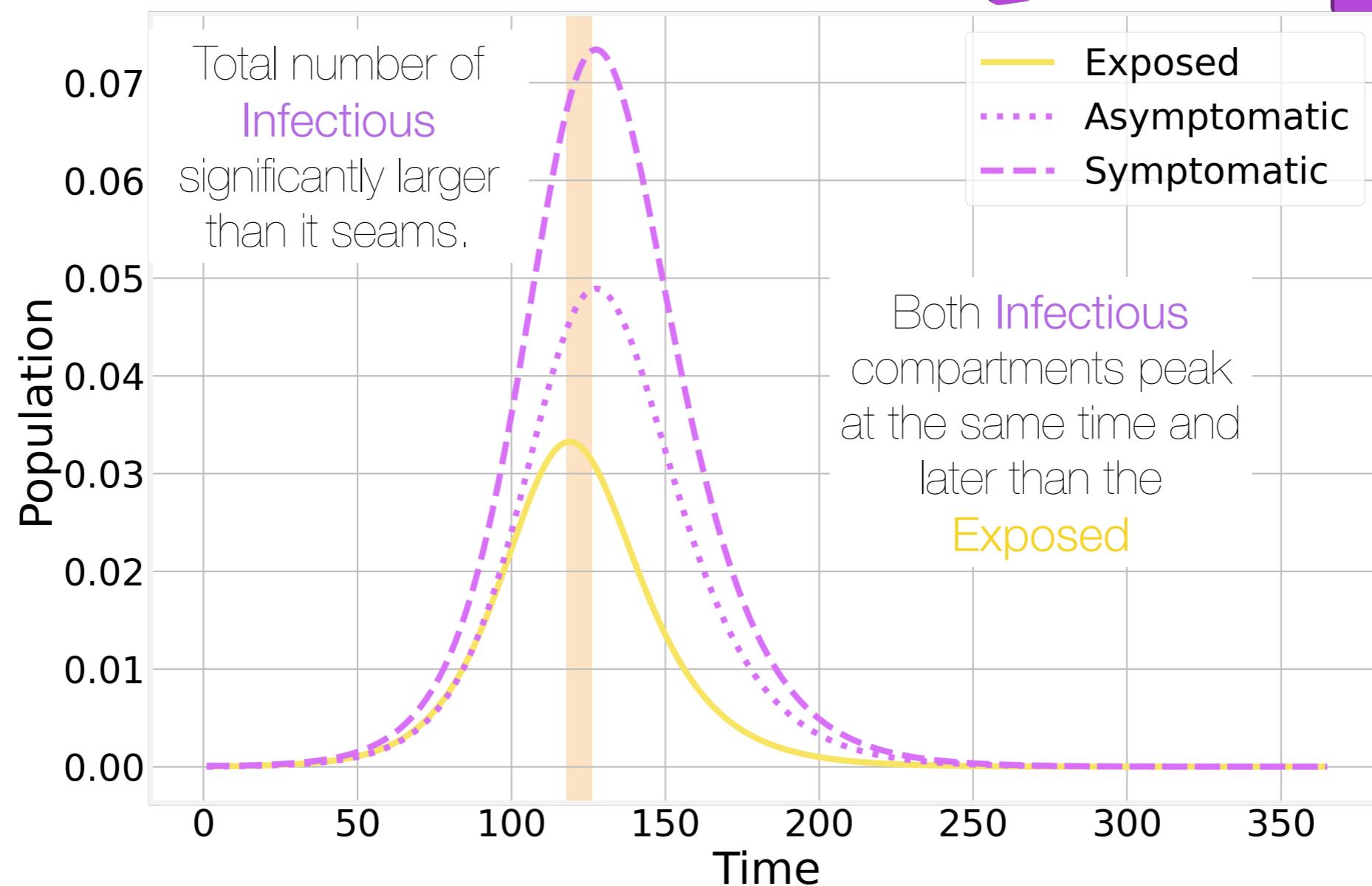
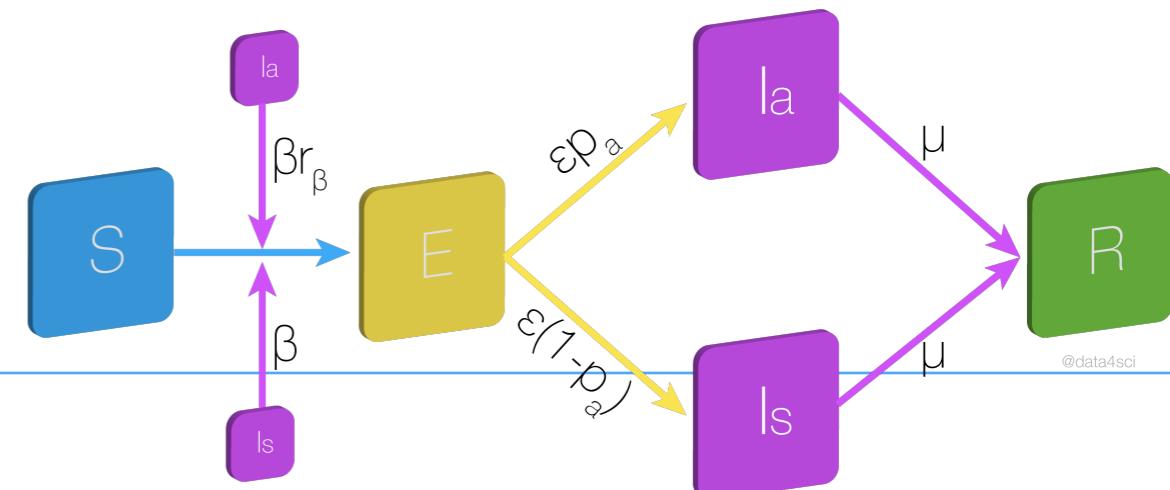
SEIRS - Waning Immunity



SEIR - Asymptomatics



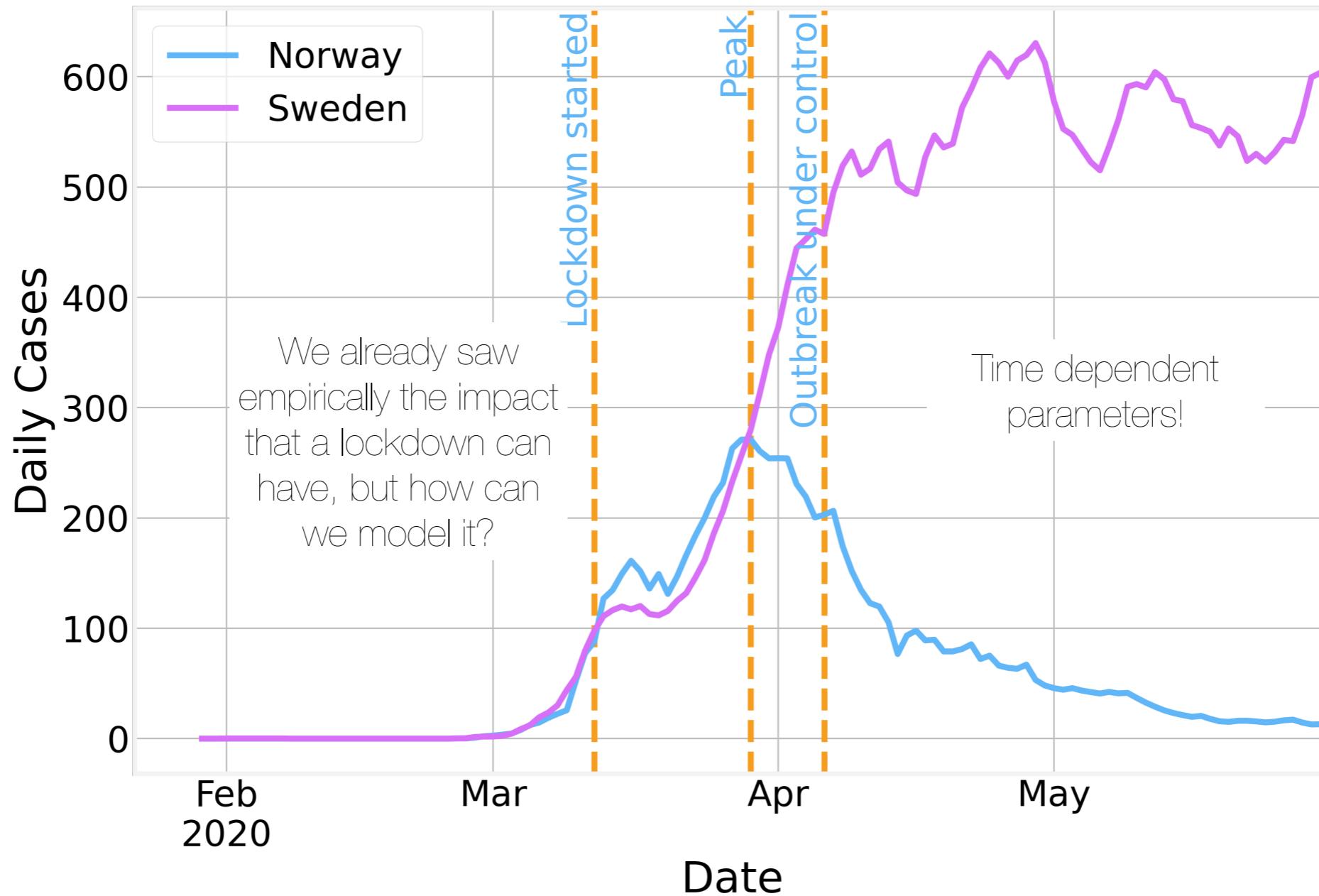
SEIR - Asymptomatics



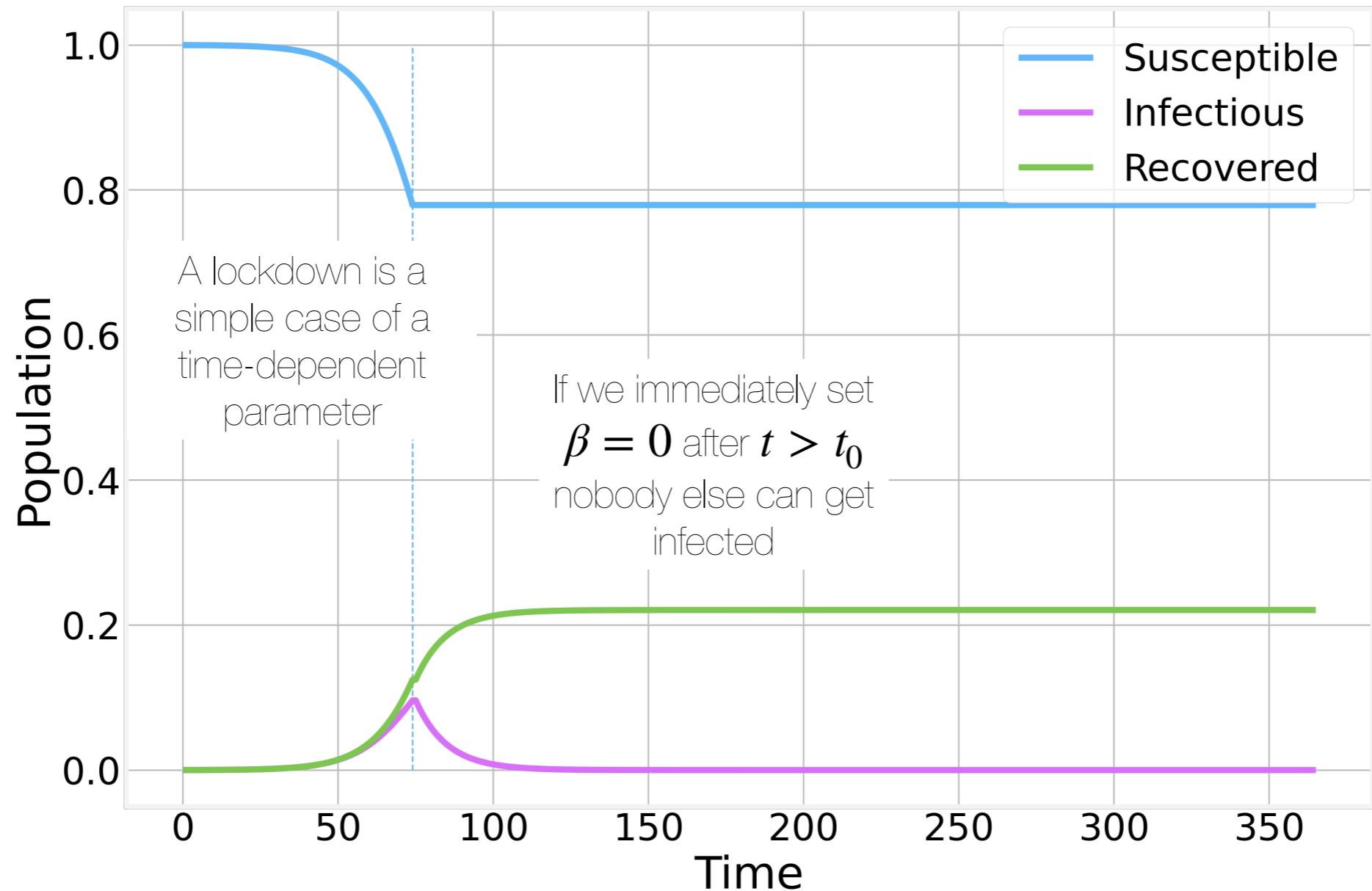


Code - Simple Models I
<https://github.com/DataForScience/CoVID19>

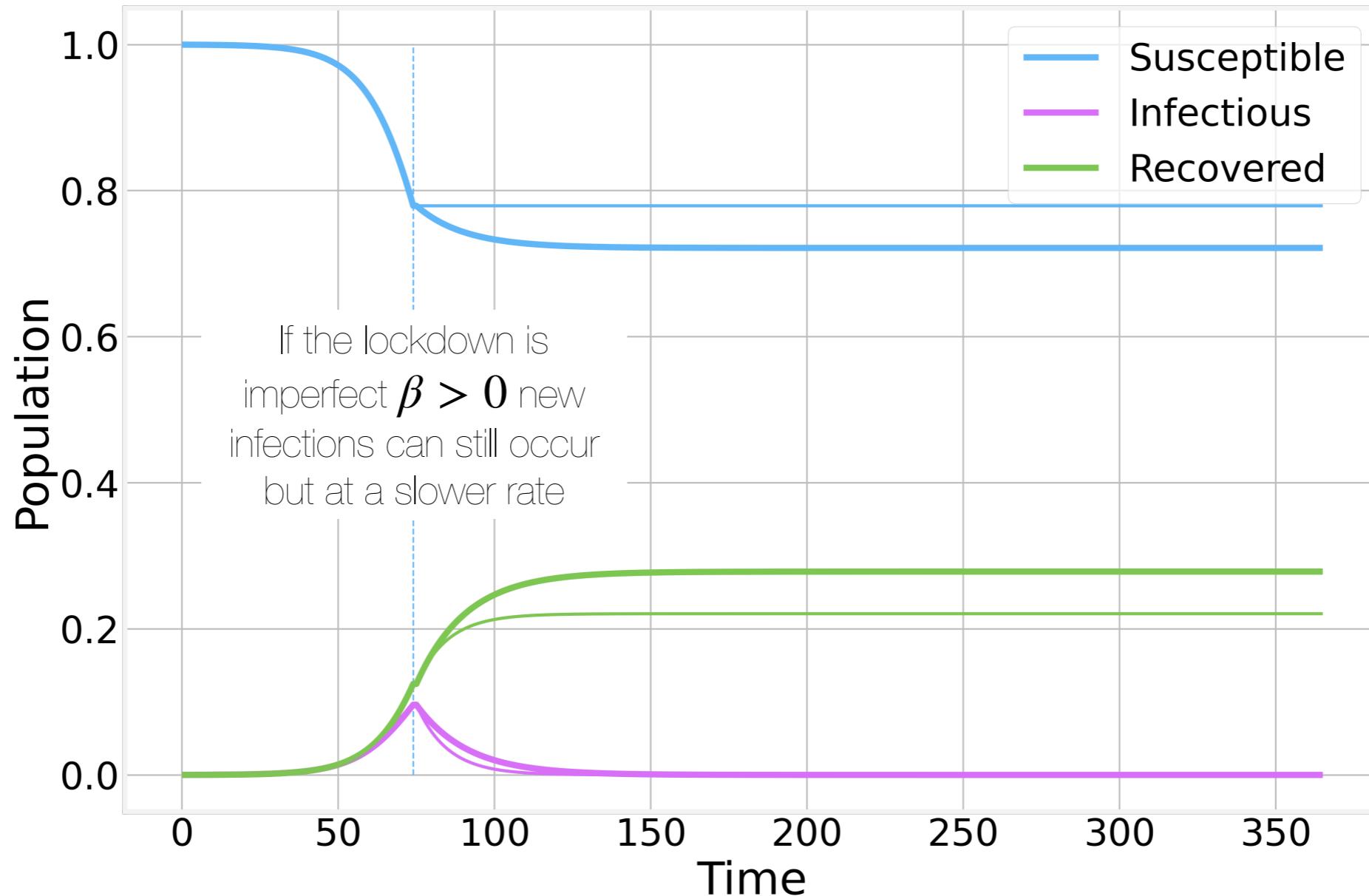
Lockdowns



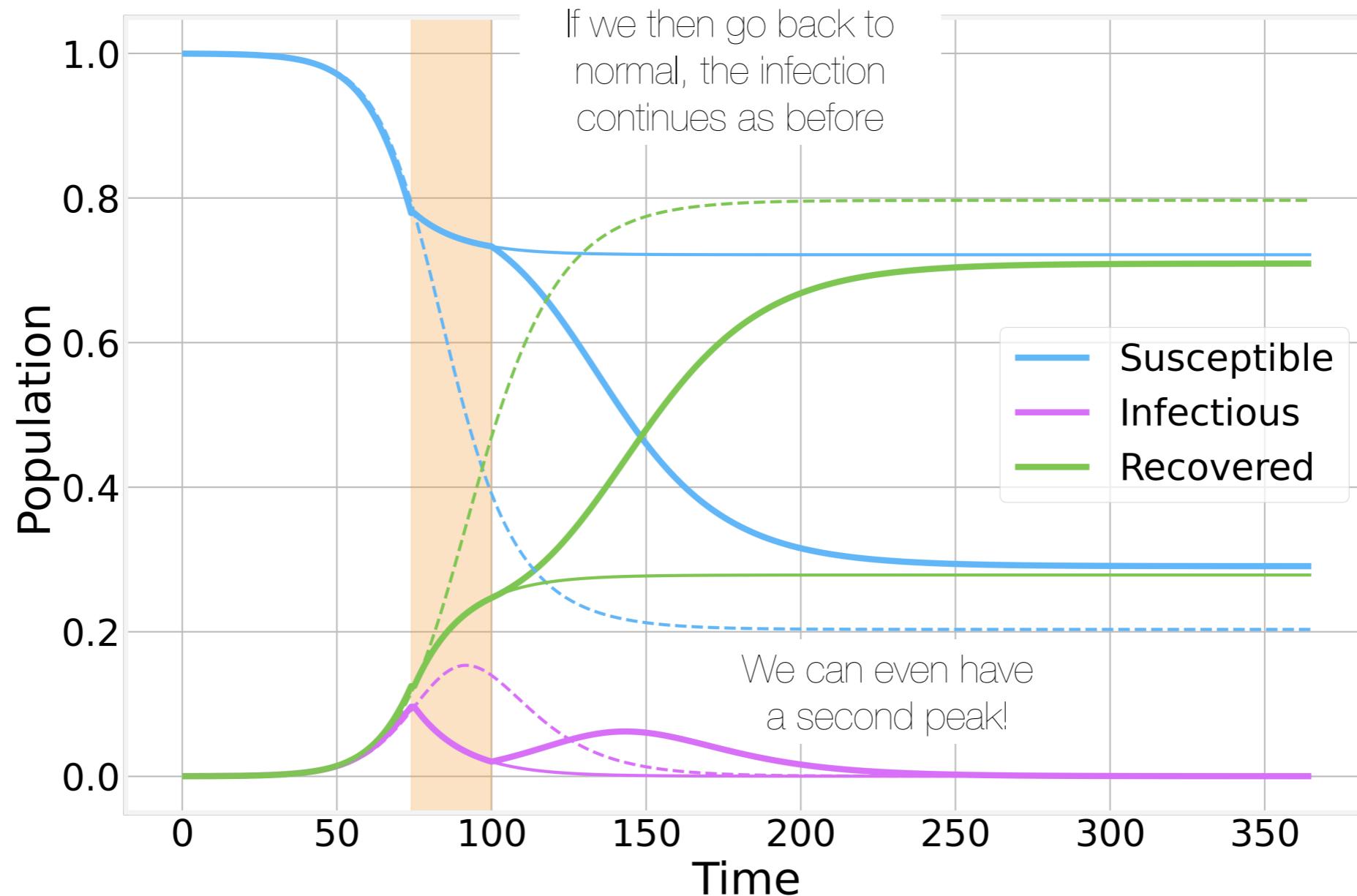
Time dependent parameters



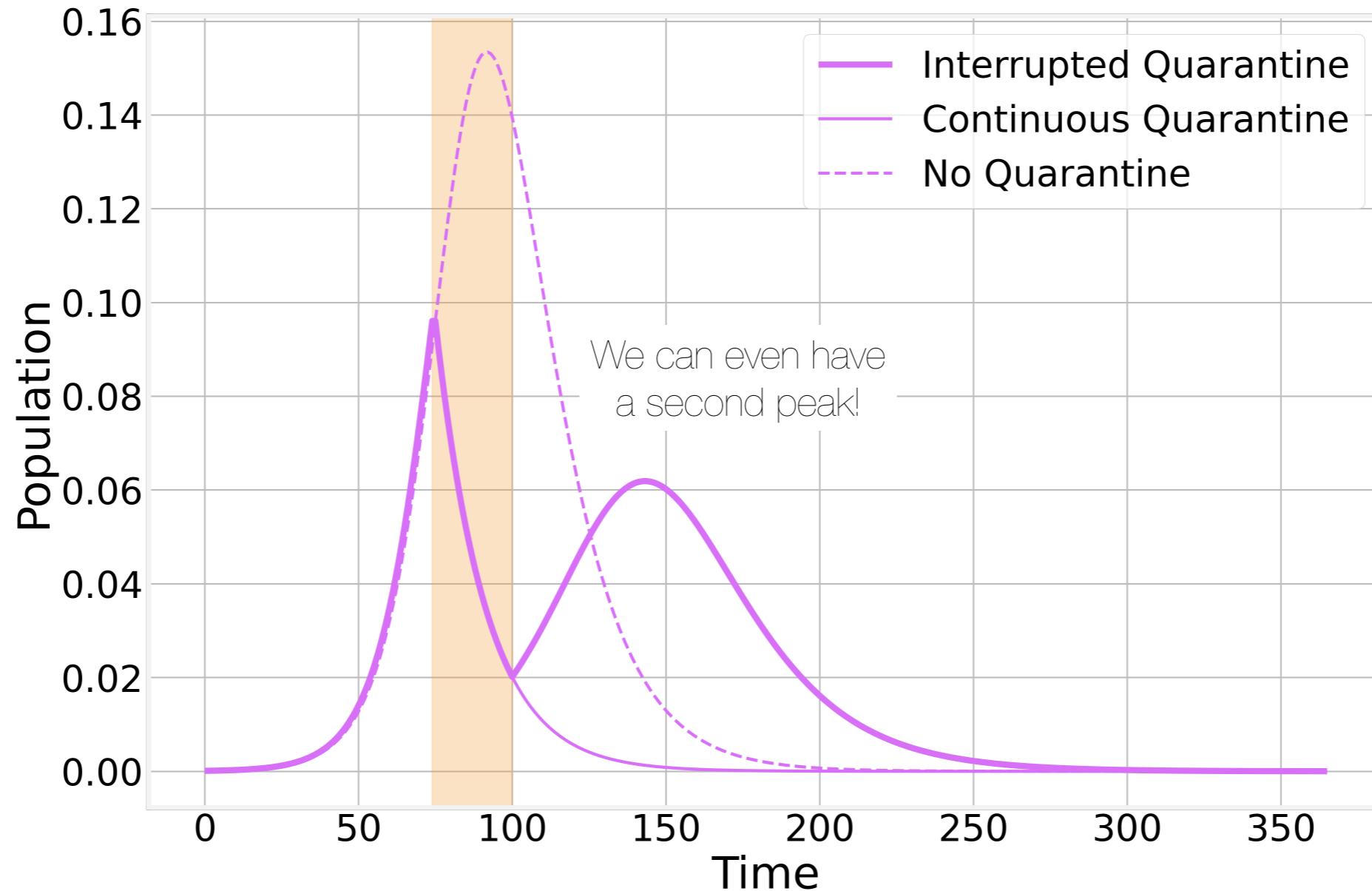
Time dependent parameters



Time dependent parameters



Time dependent parameters

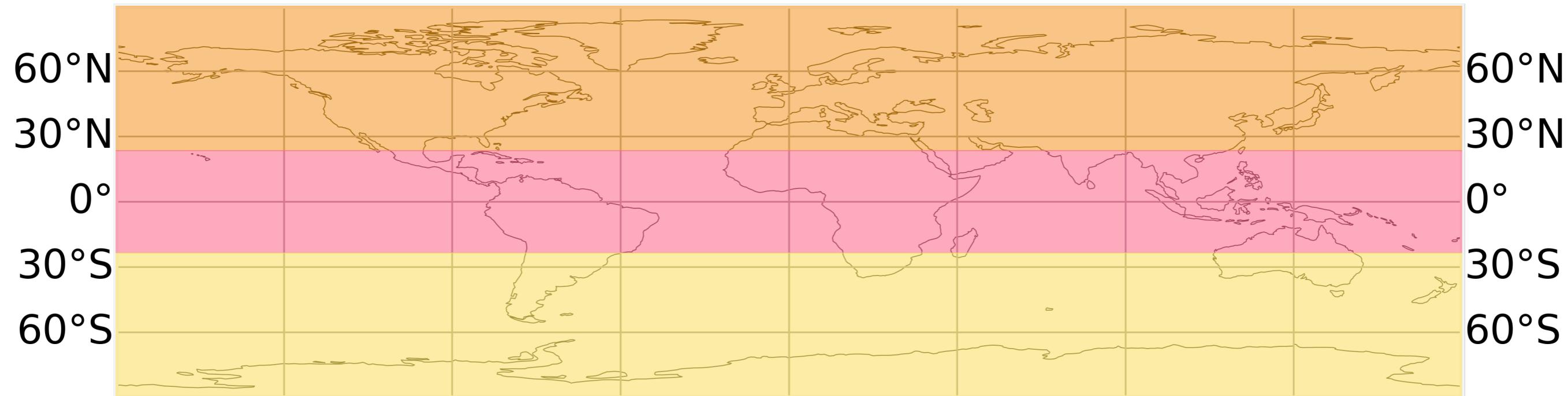




Code - Simple Models II
<https://github.com/DataForScience/CoVID19>

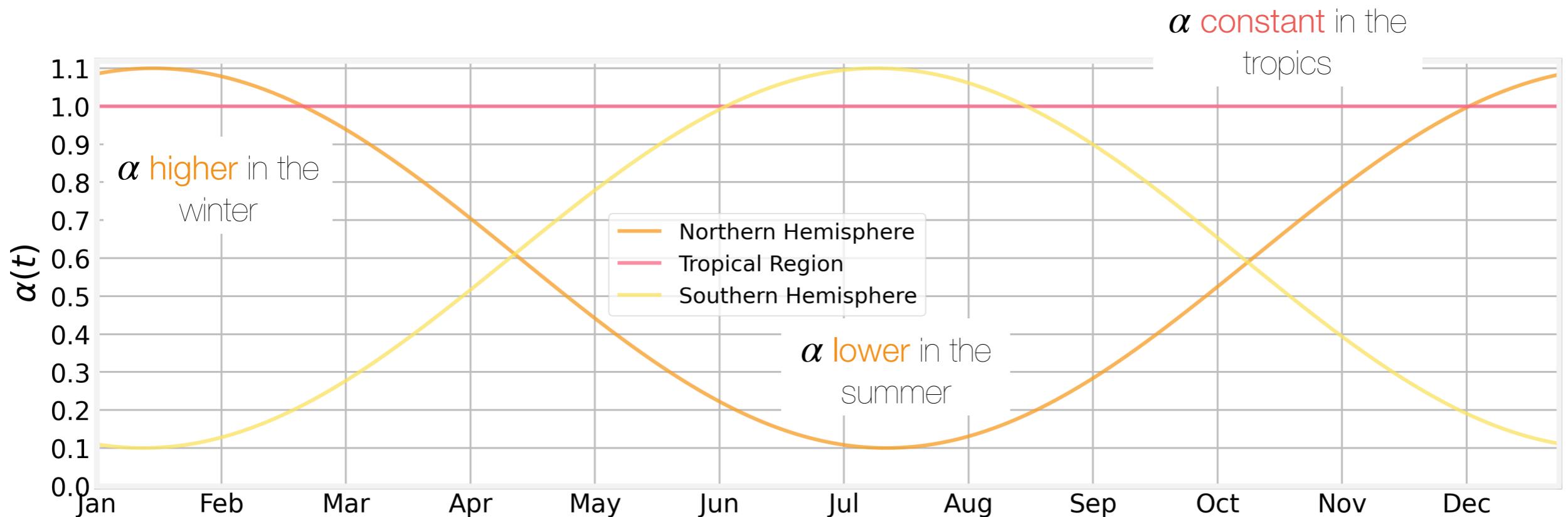
Seasonality

- Viruses that cause Influenza-like illnesses tend to be seasonal
- Seasons differ between **Northern** and **Southern** hemisphere and the **Tropics**
- This is another way in which β can vary over time

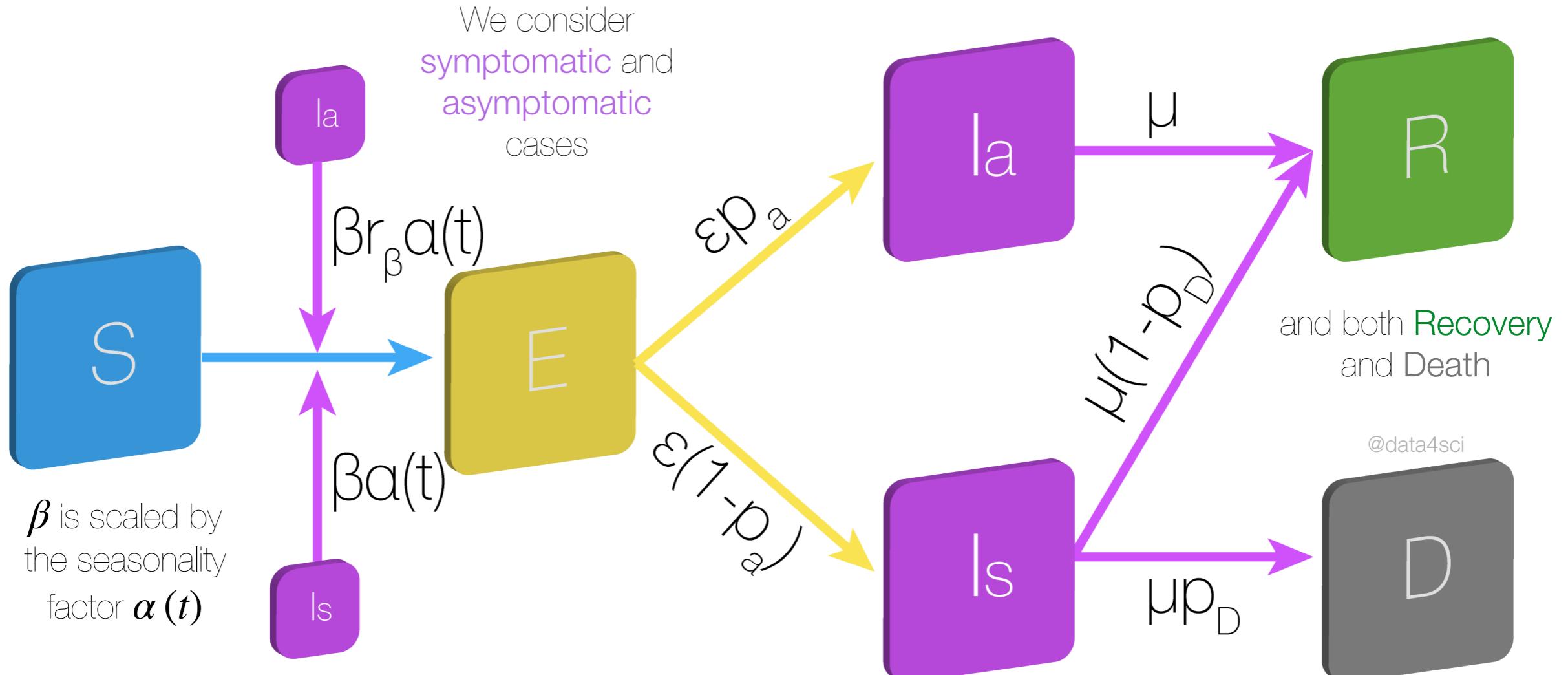


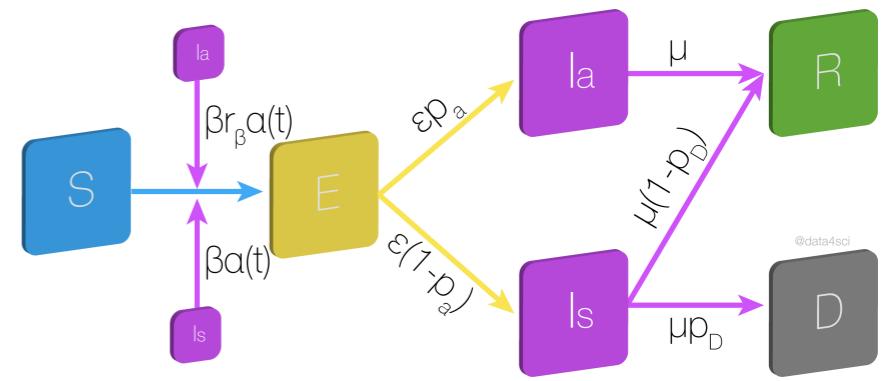
Seasonality

- Viruses that cause Influenza-like illnesses tend to be seasonal
- Seasons differ between **Northern** and **Southern** hemisphere and the **Tropics**
- This is another way in which β can vary over time: $\beta(t) = \beta_0 \alpha(t)$

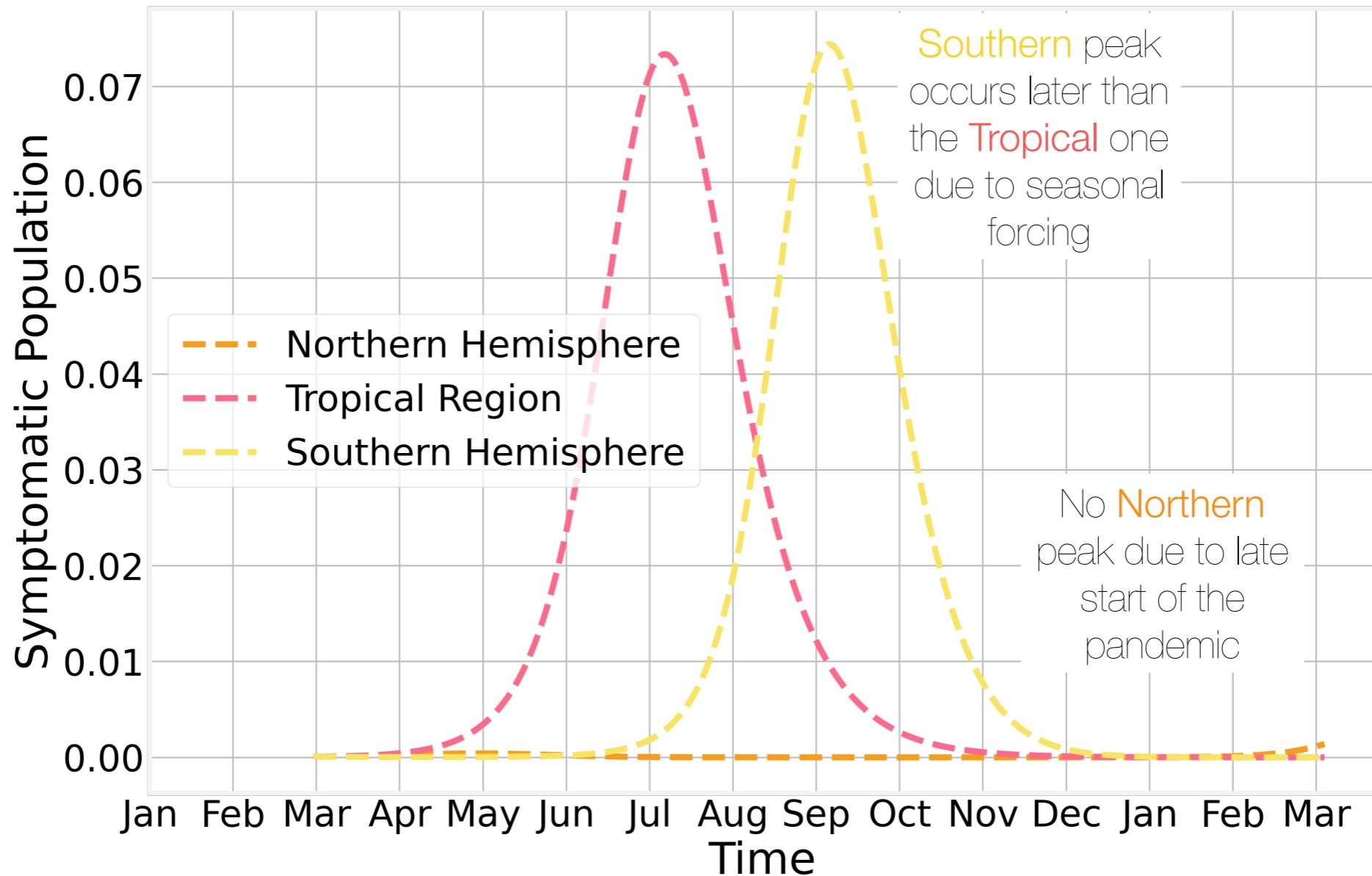


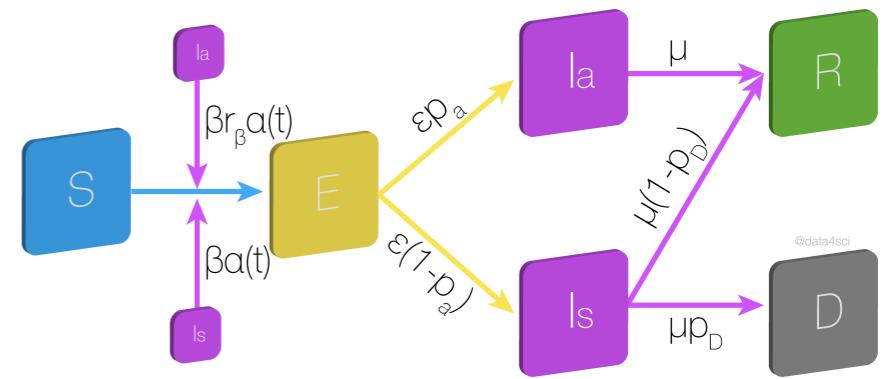
Seasonal CoVID-19 Model



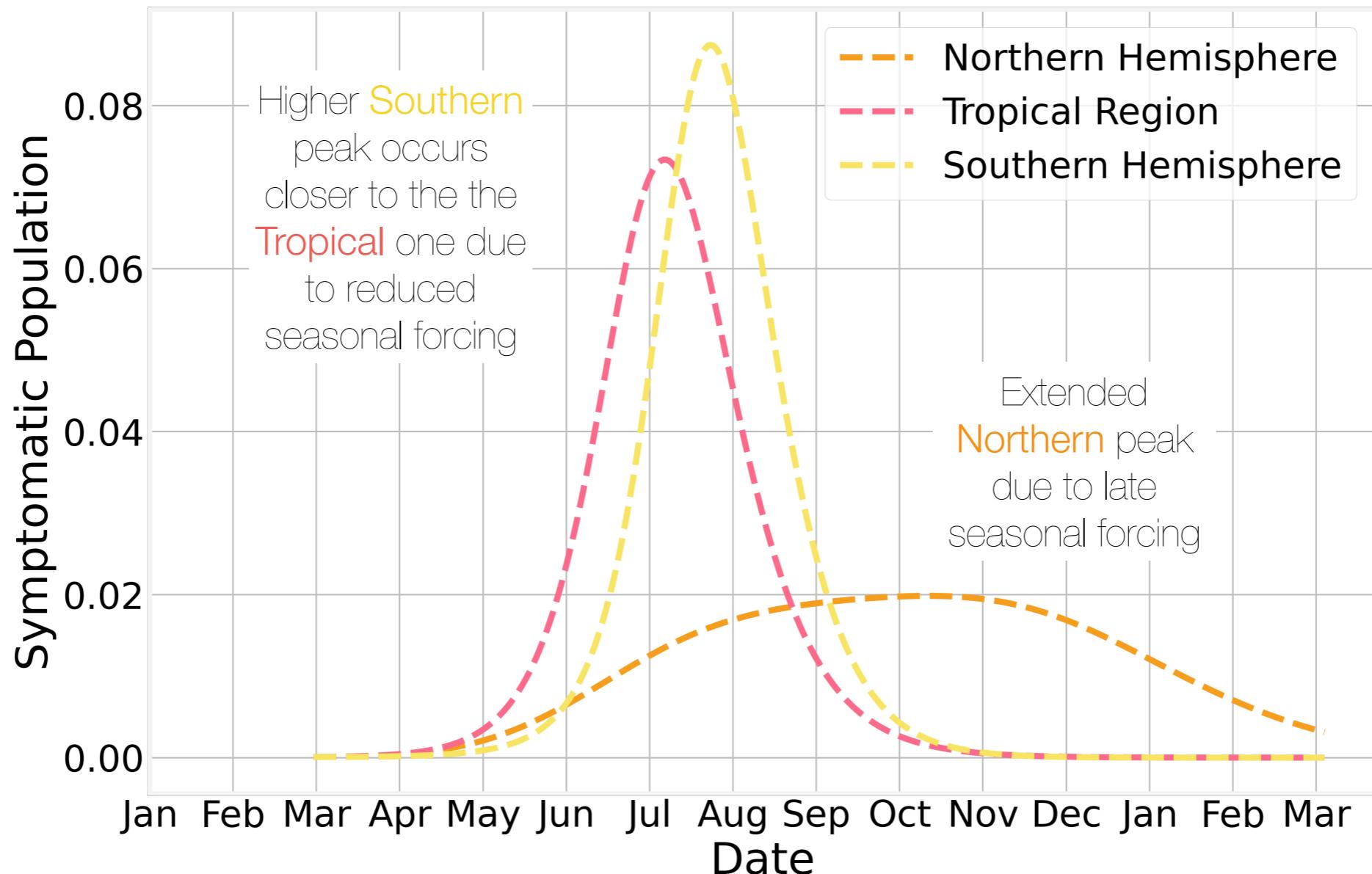


Normal Seasonality

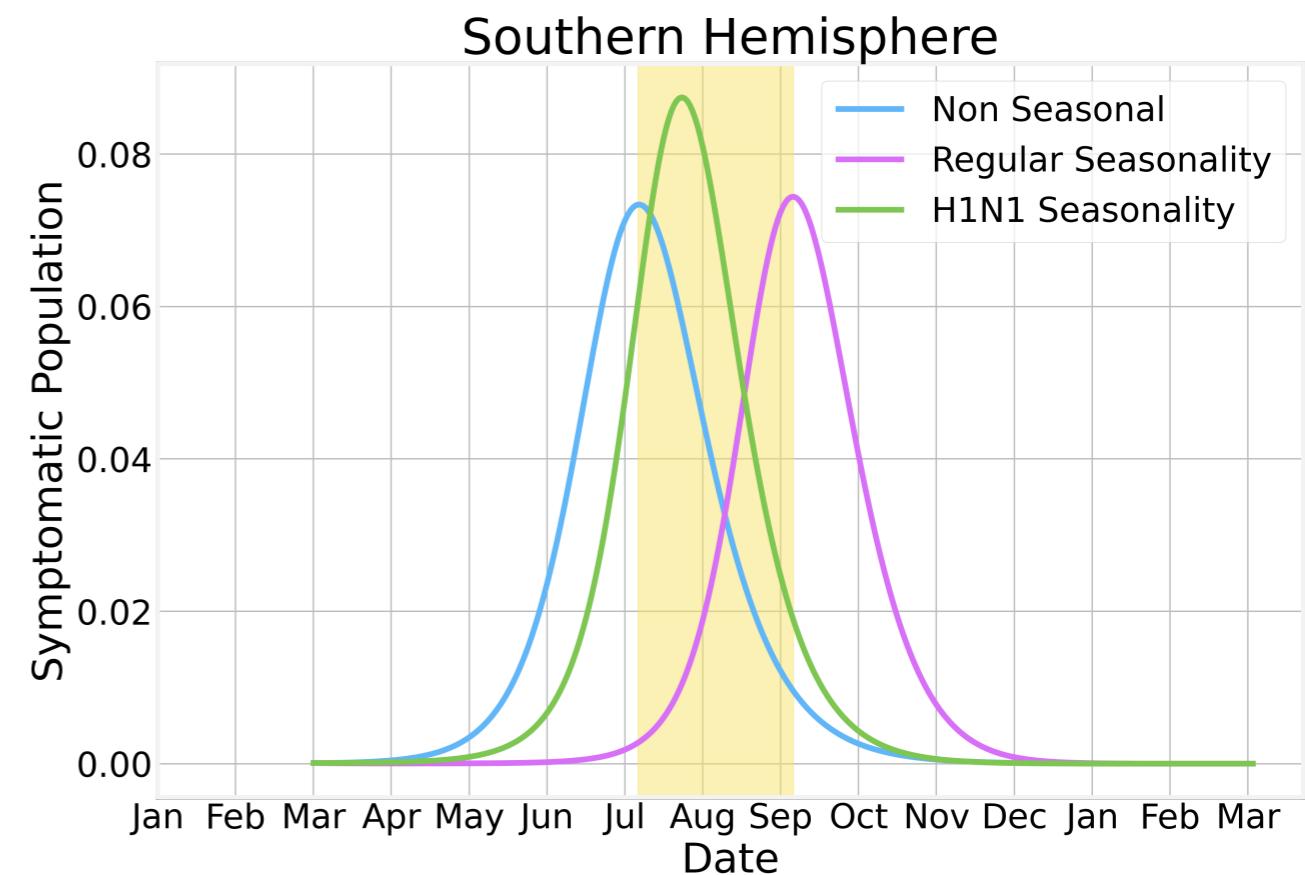
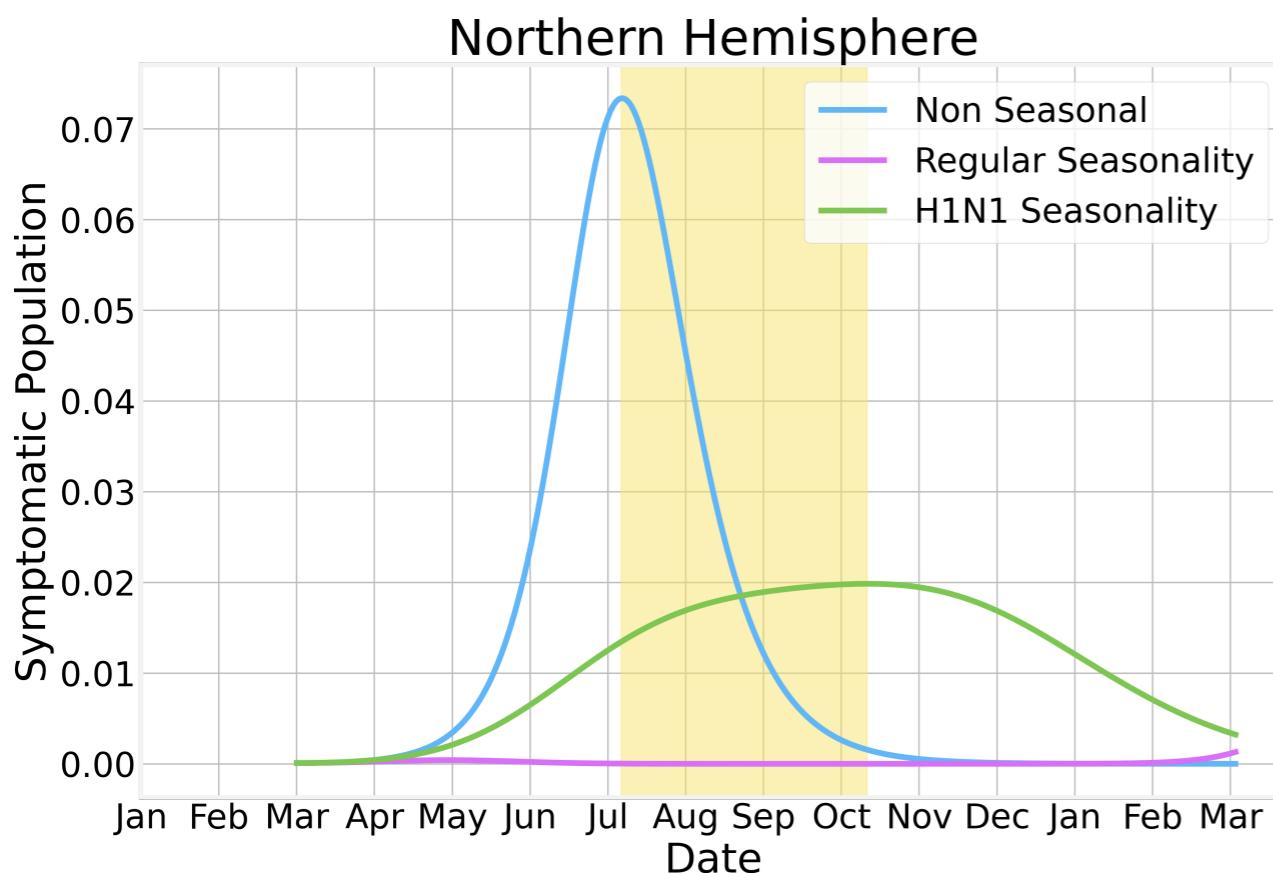




Reduced Seasonality



Direct Comparison

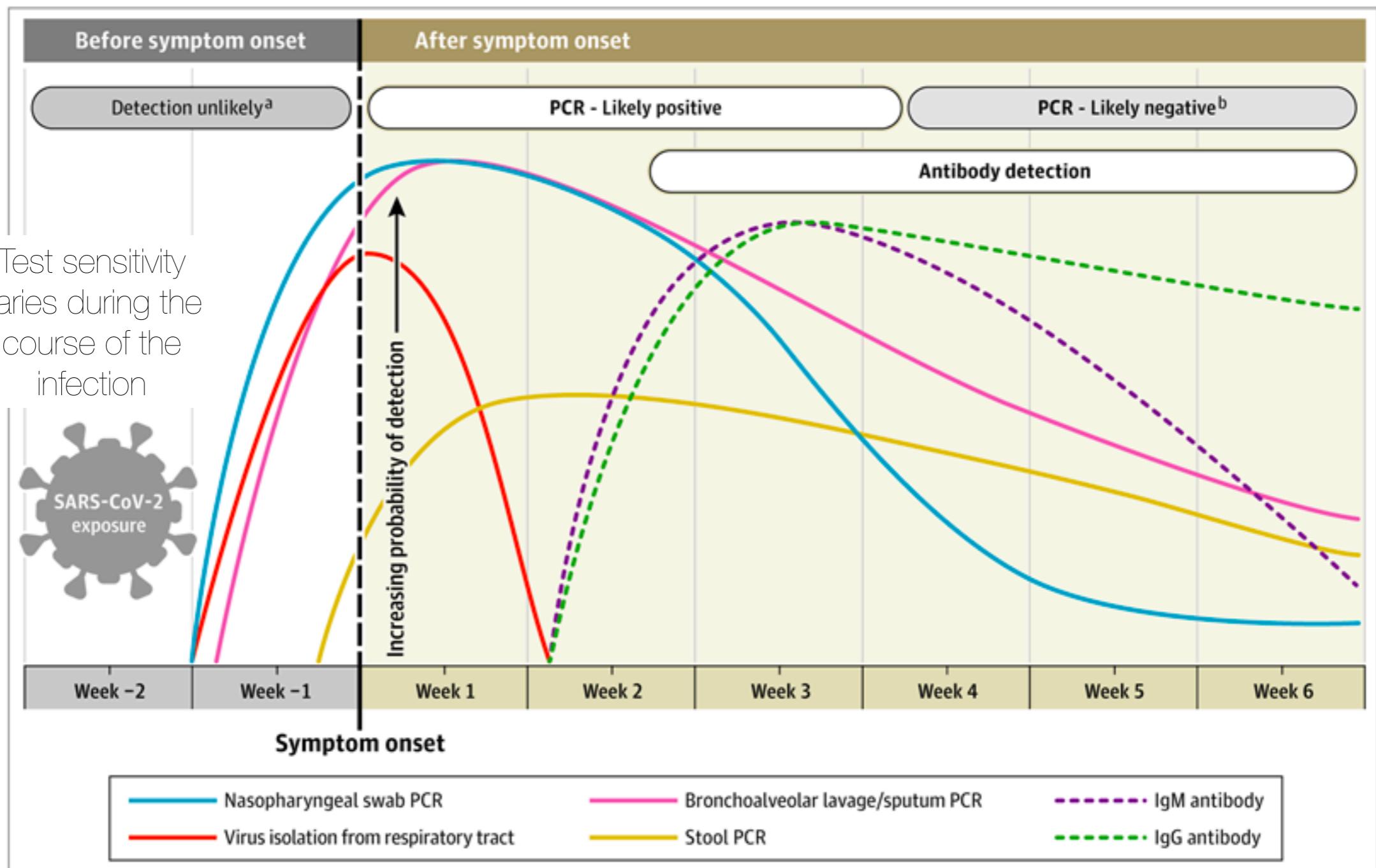


Shaded area
highlights the delay in
peak between the
various scenarios

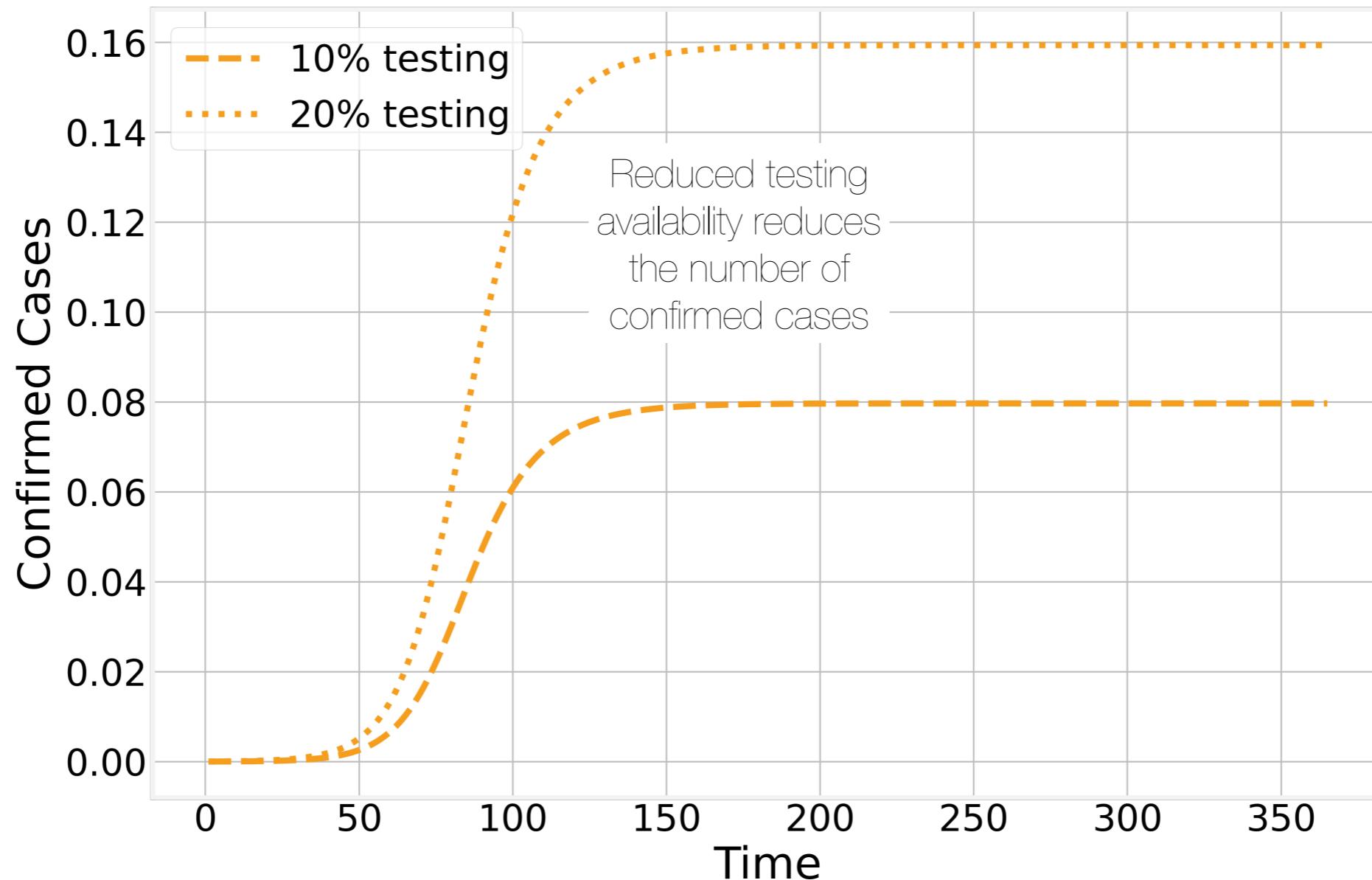


Code - Seasonality
<https://github.com/DataForScience/CoVID19>

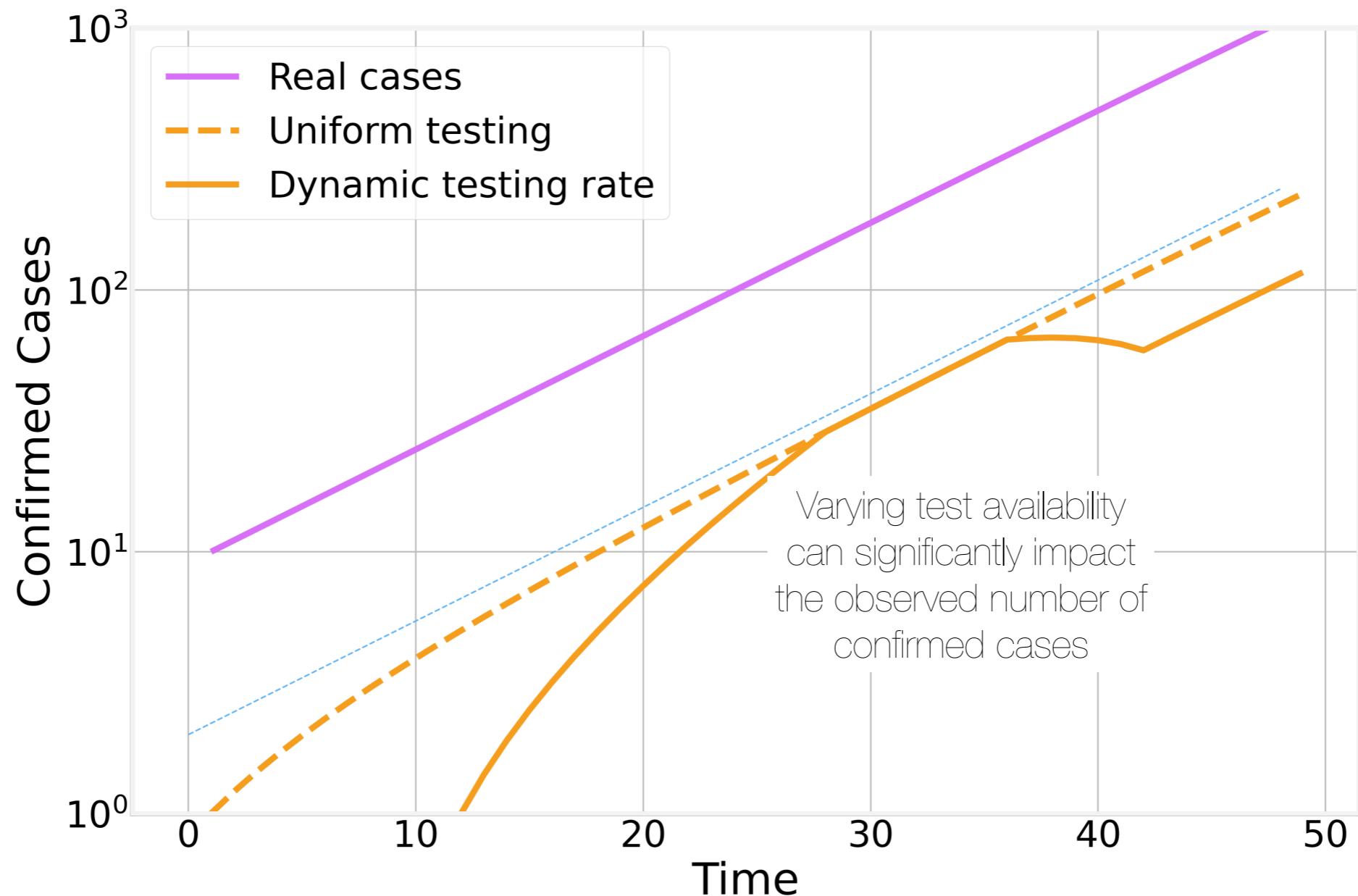
The Effects of Testing



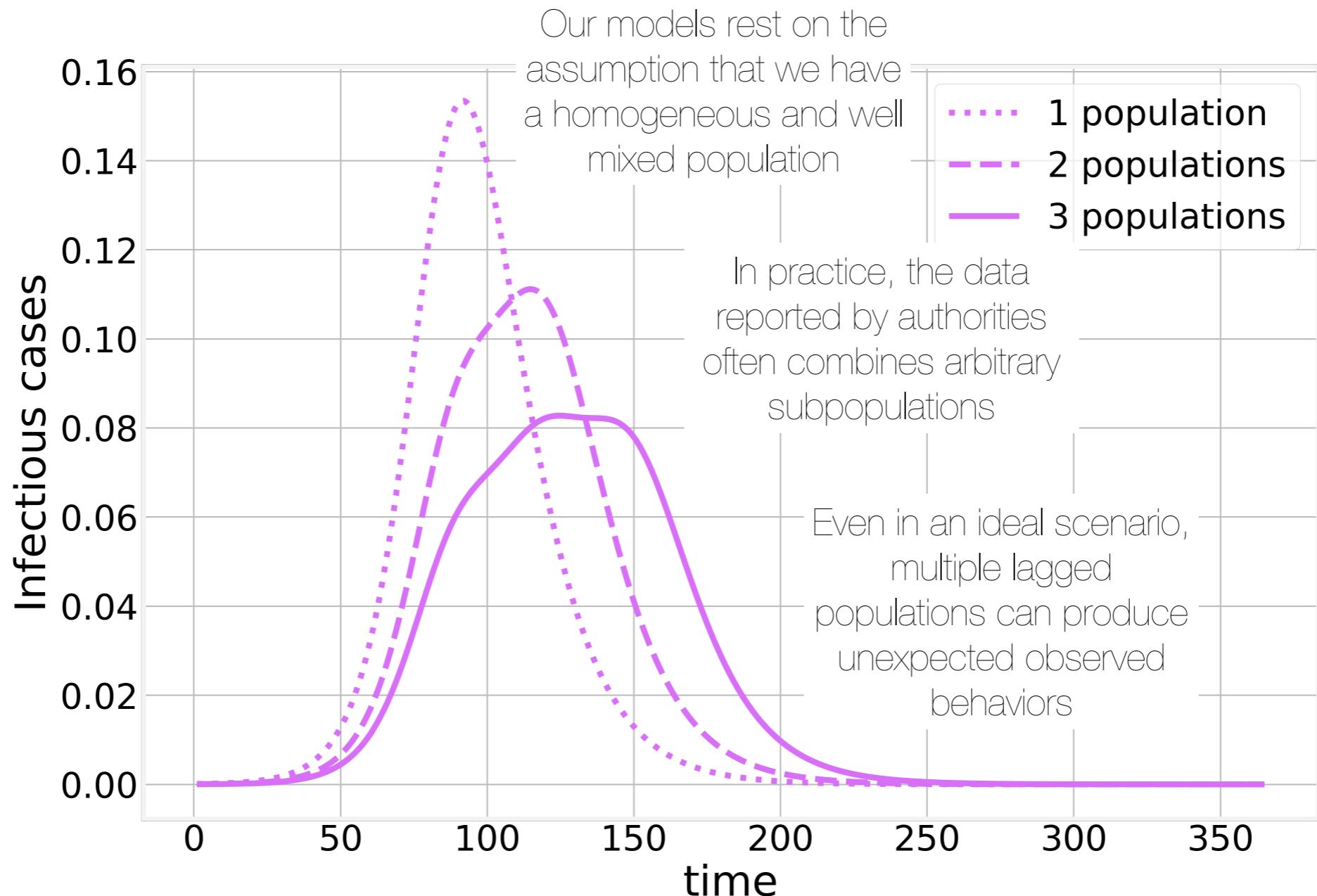
The Effects of Testing



Non-Uniform Testing



Multiple Populations





Code - Practical Considerations
<https://github.com/DataForScience/CoVID19>



3. Stochastic Models

Stochastic Models

- So far we've considered only **deterministic**, **continuous** models:
 - **Deterministic** - No room left for chance or variability (capture the average behavior)
 - **Continuous** - Using floating point values for the populations of each compartment
- However, we often need to explore the natural variability inherent in the epidemic process.
For this we need to use **stochastic**, **discrete** models:
 - **Stochastic** - Using random numbers and probabilities to generate variability
 - **Discrete** - Only allow for integer values in the population counts
- Essentially, we're moving from solving Coupled Differential Equations to running a full fledged simulation of the epidemic

Stochastic

Similar approach as in the case of numerical integration above

Multinomials distribute a **number** of elements among a set of possibilities according to their probabilities

```
for t in time:  
    pop = values[-1]  
    new_pop = values[-1].copy()  
    N = np.sum(pop)  
  
    for comp in comps:  
        trans = list(self.transitions.edges(comp, data=True))  
  
        prob = np.zeros(len(comps), dtype='float')  
  
        for _, node_j, data in trans:  
            source = pos[comp]  
            target = pos[node_j]  
  
            rate = data['rate']  
  
            if 'agent' in data:  
                agent = pos[data['agent']]  
                rate *= pop[agent]/N  
  
            prob[target] = rate  
  
        prob[source] = 1-np.sum(prob)  
  
        delta = random.multinomial(pop[source], prob)  
        delta[source] = 0  
  
        changes = np.sum(delta)  
  
        if changes == 0:  
            continue  
  
        new_pop[source] -= changes  
  
        for i in range(len(delta)):  
            new_pop[i] += delta[i]  
  
    values.append(new_pop)
```

Rate for spontaneous transactions

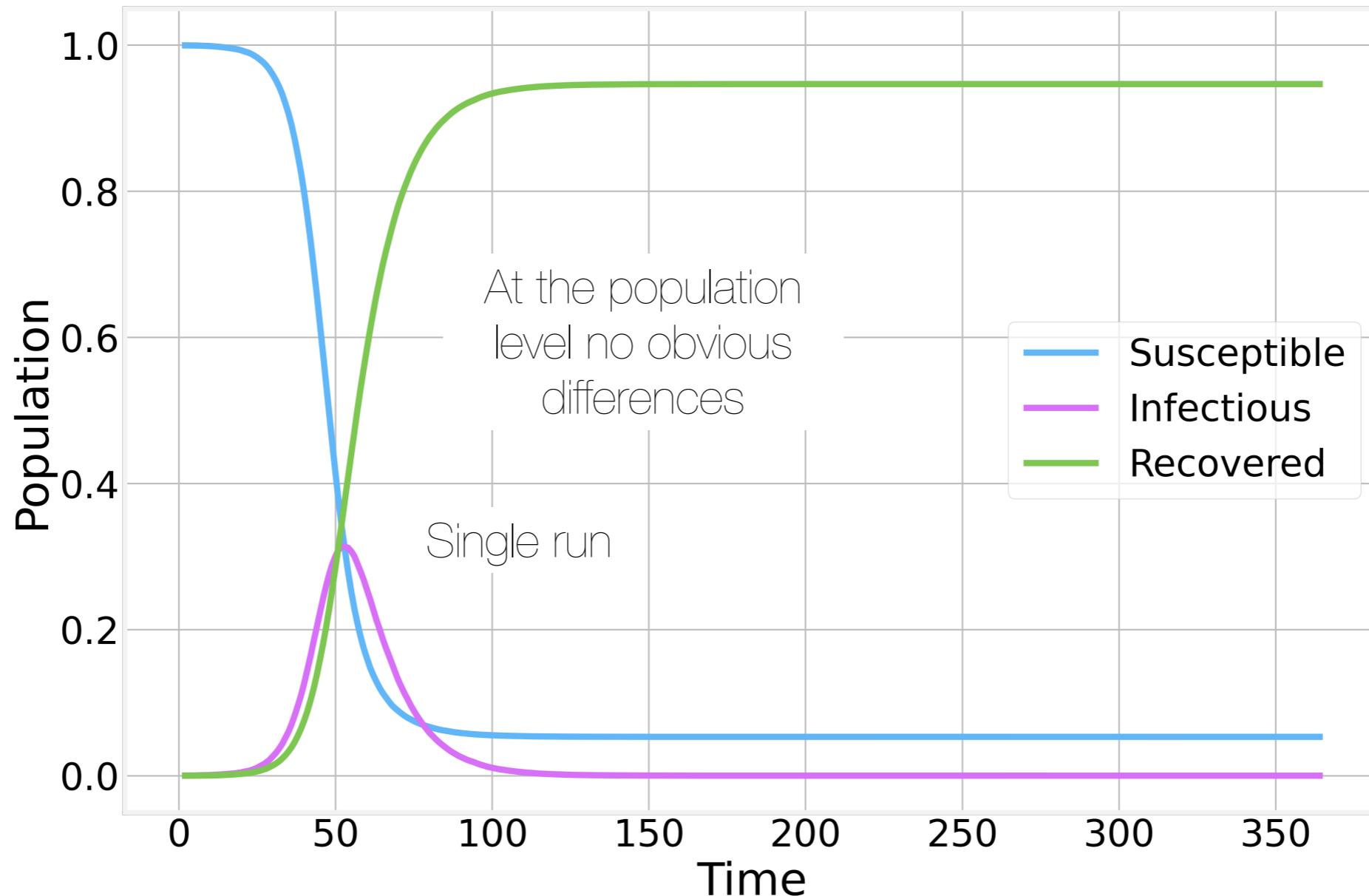
Modify for interactive transaction

Update the transition probabilities

Update the compartment populations

EpiModel.py

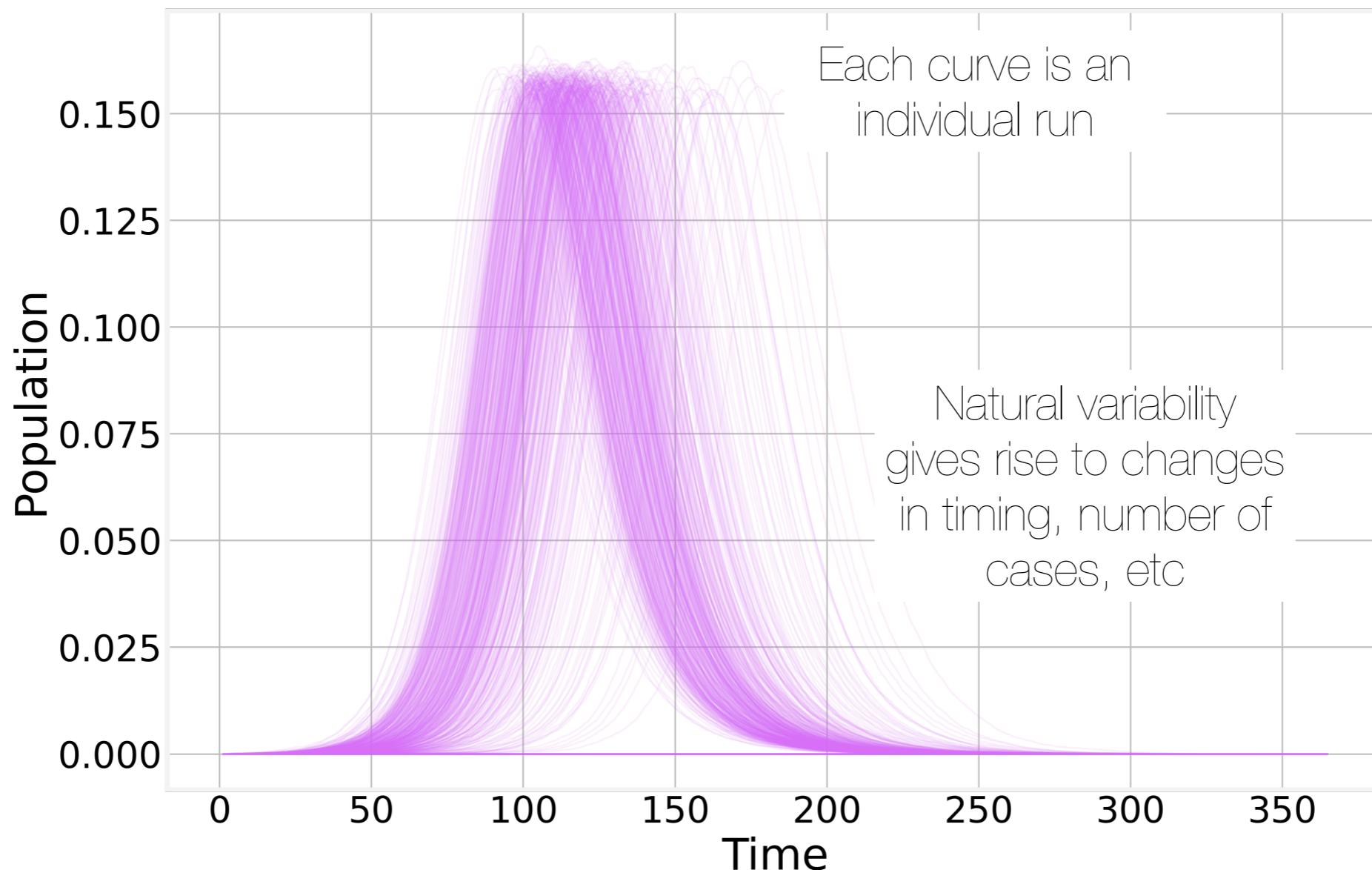
Stochastic Model



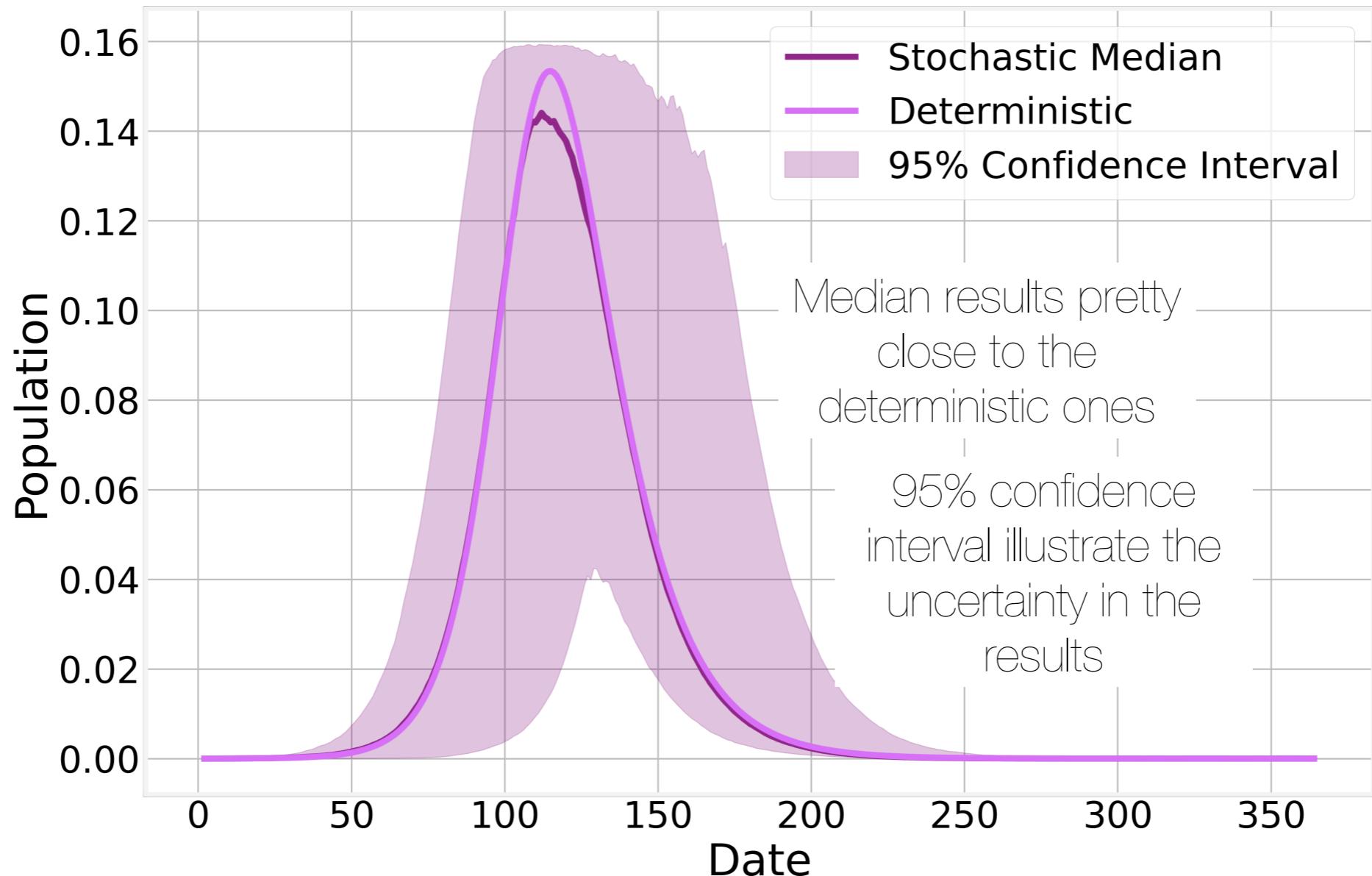
Stochastic Model



Stochastic Model

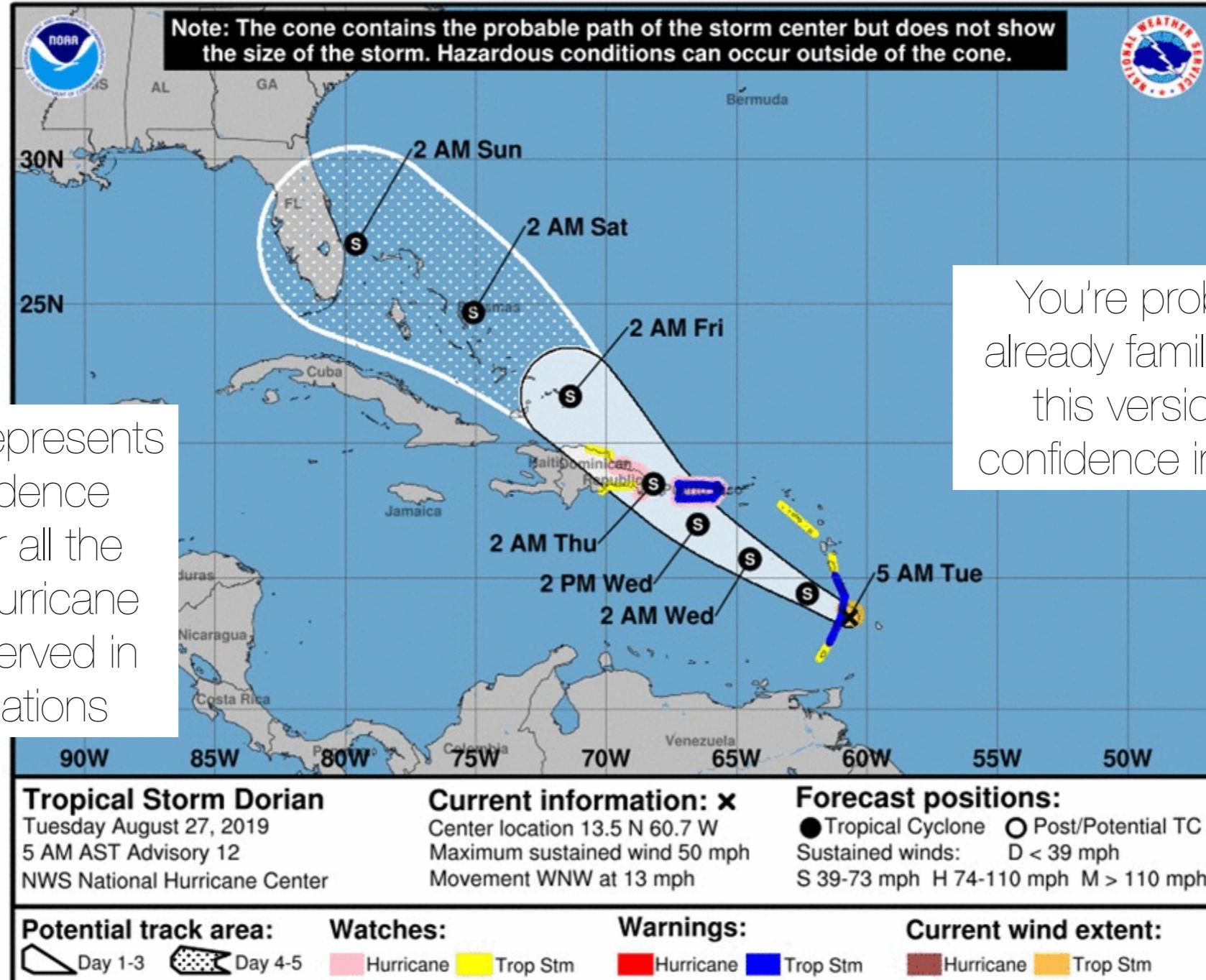


Stochastic Model

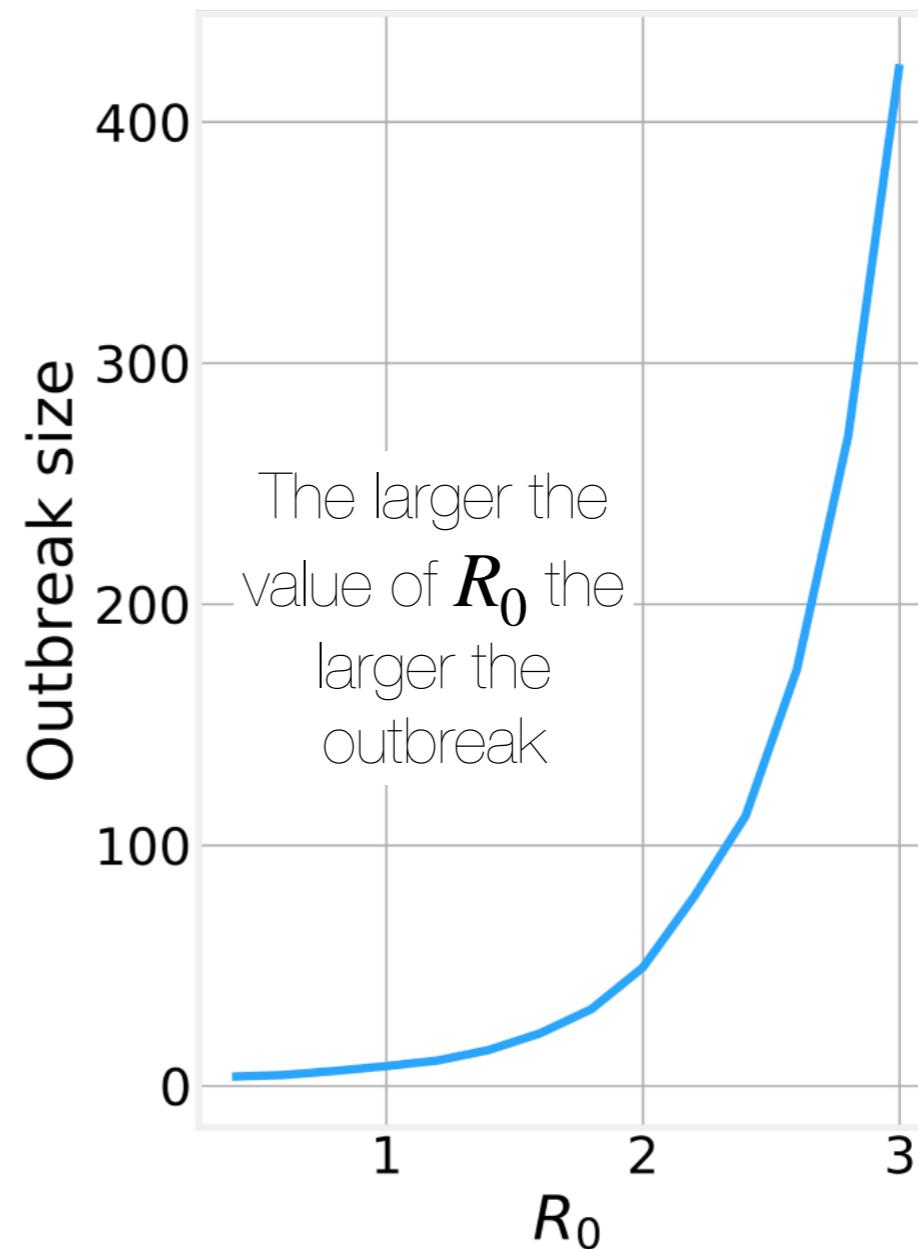
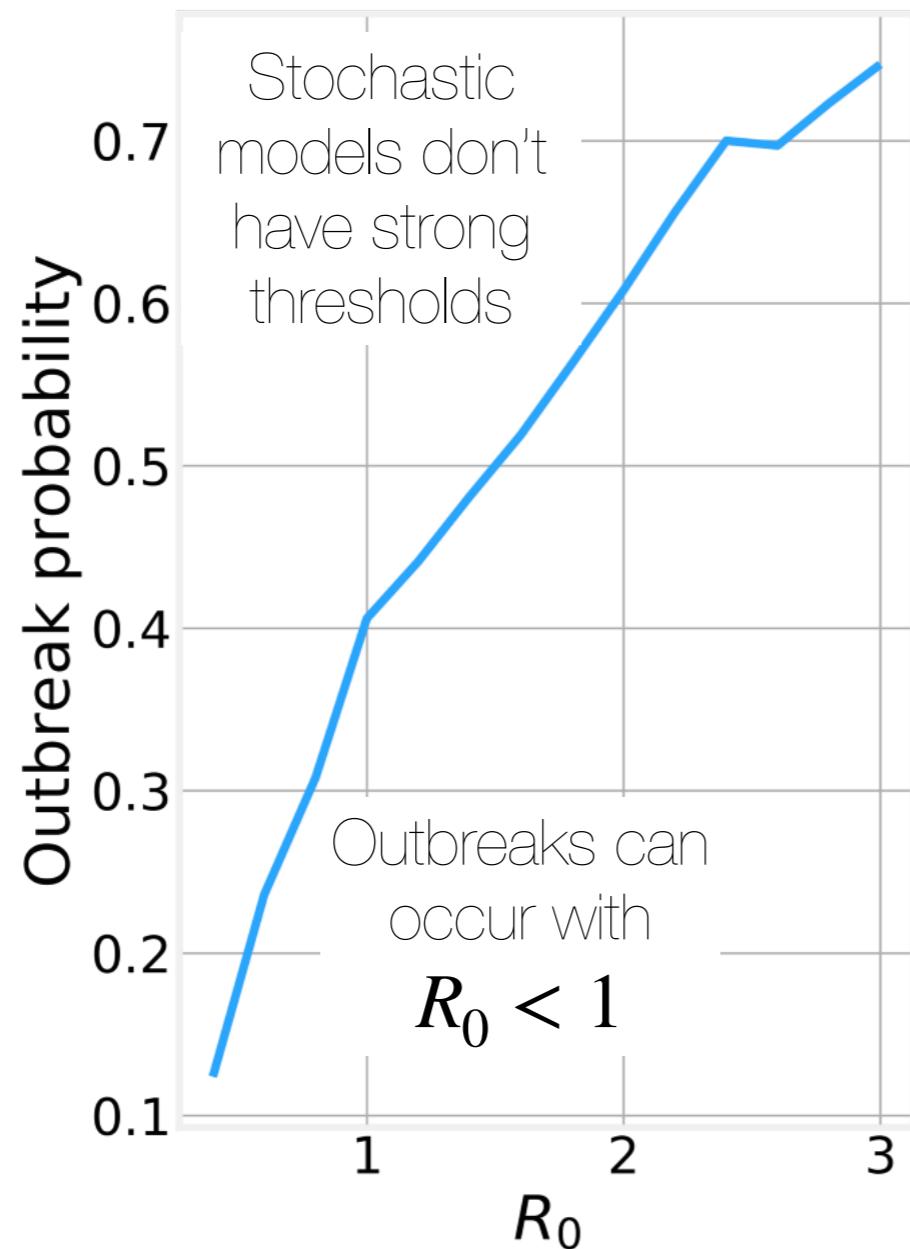


Stochastic Model

The cone represents the confidence interval for all the possible hurricane paths observed in the simulations



Initial uncertainty





Code - Stochastic Models
<https://github.com/DataForScience/CoVID19>

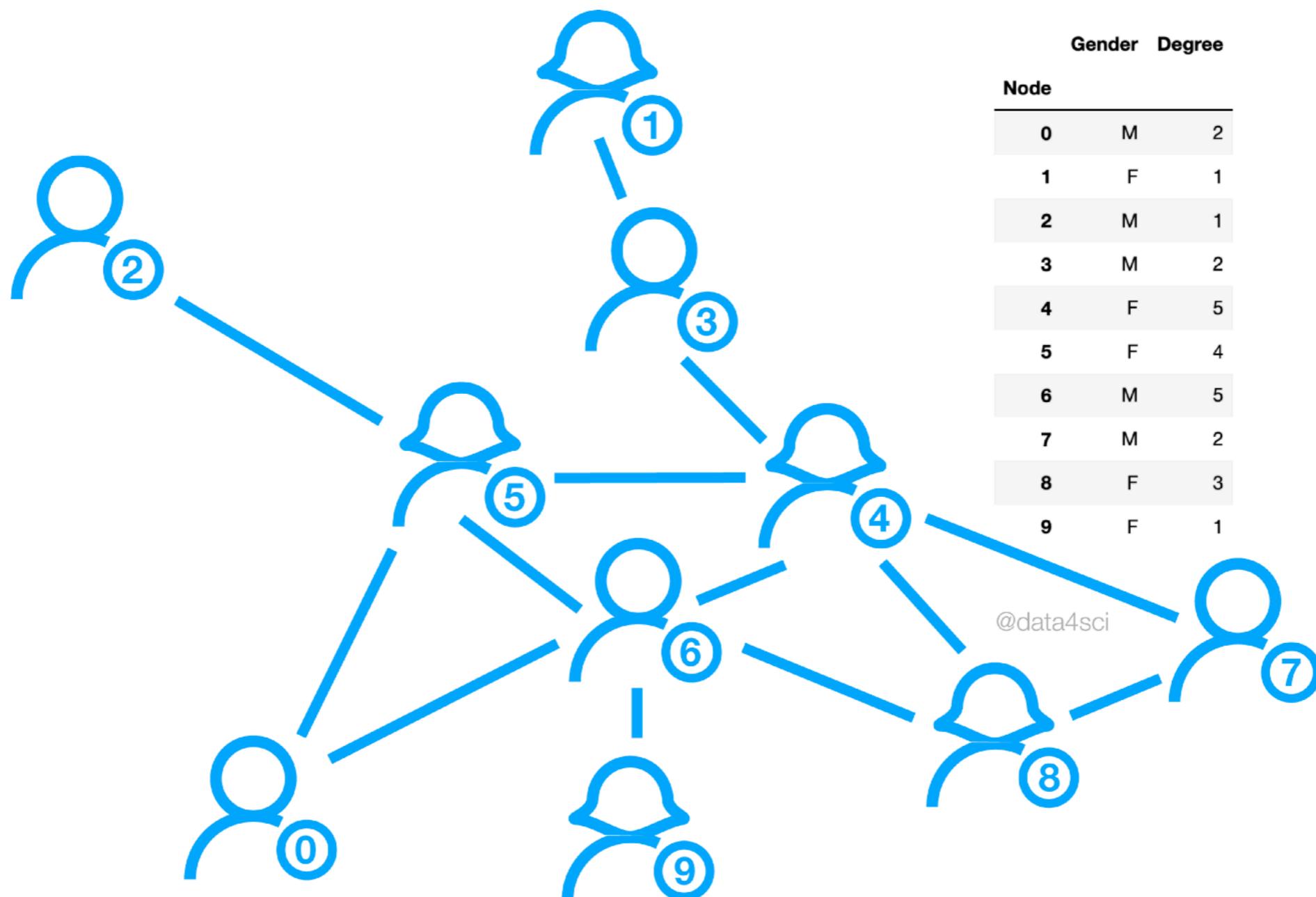


4. Network Models

Networks and Graphs

- **Network**: A system that is constituted by nodes (subcomponents) interconnected by links (connections, relationships, etc)
- Network vs Graphs: Roughly equivalent. "Network" is typically preferred by Physicists and Engineers while "Graph" is more common among Mathematicians.
- Mathematically, a network is a set of vertices V (nodes) and edges E (links).
- A **link** is a pair of nodes, $e = (v, v') \in E$
- In the case of undirected networks, the order of v and v' does not matter. In the case of directed networks, (v, v') is a link from v to v' .
If $(v, v') \in E$ and $(v', v) \in E$ the two nodes are reciprocally connected.
- In the case of weighted networks, links are also assigned with a weight function, characterizing the importance or weight of the link.

Social Networks



Facebook



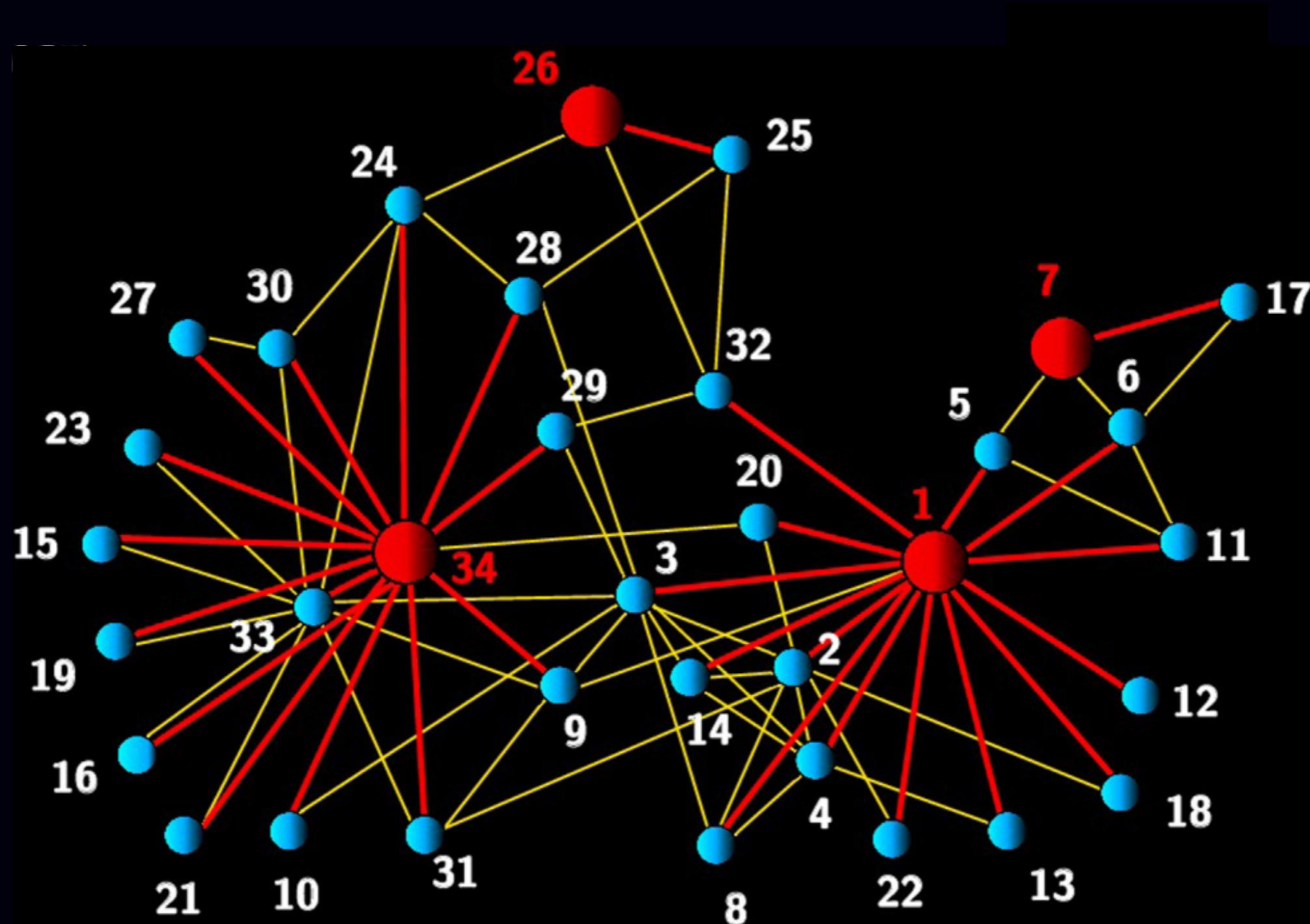
facebook

December 2010

@bgoncalves

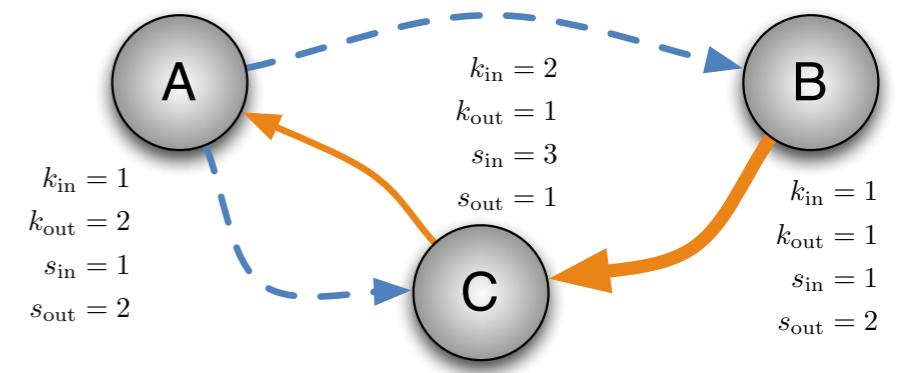
www.data4sci.com

Zachary Karate Club



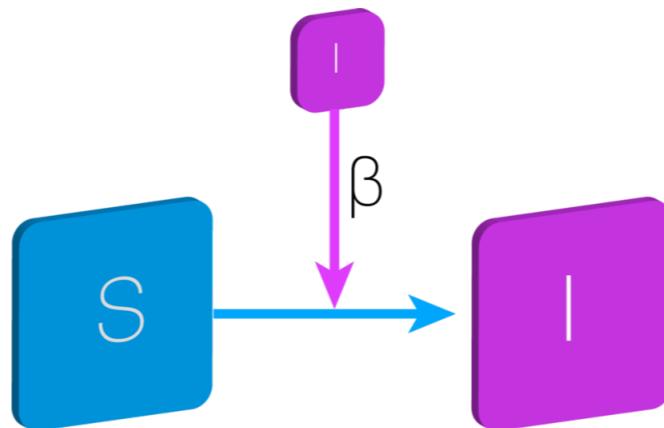
Mathematical Details

- Mathematical object, with set of nodes and edges
- **Node** - Individual Element
- **Edge** - Connection between element
- **Degree** - Number of edges connected to a node
- **Weighted Edge** - Edge with a weight associated
- **Directed** Edge - "One way street"

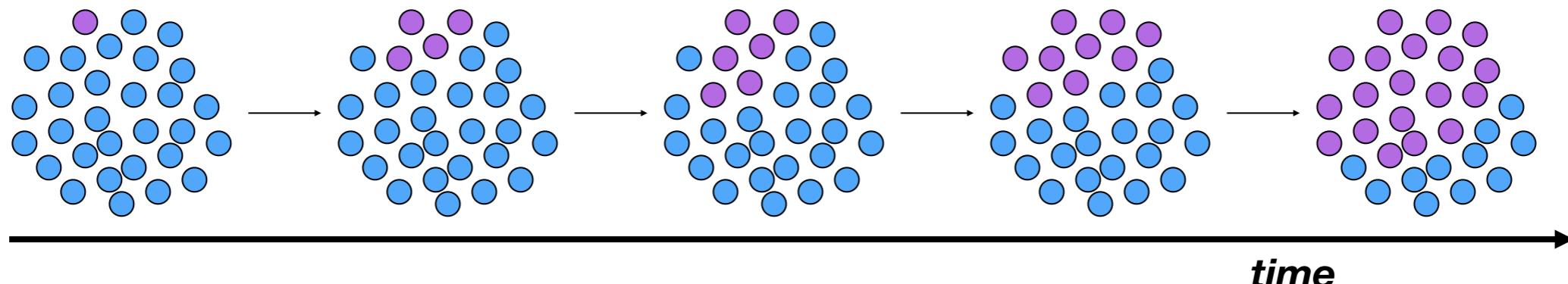


Network Spreading

- Let us consider the simplest example, the SI model:



- As before, eventually everyone becomes infected:



- But since spreading can only occur along the edges of the network, the **network topology** will determine how fast or how slow the disease will spread

Network Spreading

- In this case, we can't just numerically integrate a set of differential equations, we must simulate the process.
- We model each node as if it were an individual agent who has to make decisions at each time step
- At each time step, we have to:
 - go through all nodes
 - if the node is infectious, see if it infects one of its neighbors
 - check if any spontaneous transactions occur

Interacting Transitions

```
if state_i in infections:  
    NN = list(self.network.neighbors(node_i))  
    np.random.shuffle(NN)  
  
    for node_j in NN:  
        state_j = population[t-1, node_j]  
  
        if state_j in infections[state_i]:  
            prob = np.random.random()  
  
            if prob < infections[state_i][state_j]['rate']:  
                new_state = infections[state_i][state_j]['target']  
                population[t, node_j] = new_state  
  
                if new_state in active_states:  
                    active_nodes.add(node_j)
```

Generate a random number

Go through each neighbor in random order

If the number is smaller than the infection probability

Infect the neighbor

NetworkEpiModel.py

@bgoncalves

Spontaneous Transitions

```
if state_i in self.spontaneous:  
    n_trans = len(self.spontaneous[state_i])  
  
    prob = np.zeros(len(pos))  
  
    for target in self.spontaneous[state_i]:  
        prob[pos[target]] = self.spontaneous[state_i][target]  
  
    prob[pos[state_i]] = 1-np.sum(prob)  
  
    new_state = comps[np.argmax(random.multinomial(1, prob))]  
  
    if new_state != state_i:  
        population[t, node_i] = new_state  
  
        active_nodes.add(node_i)  
  
        if new_state not in active_states:  
            active_nodes.remove(node_i)  
  
    continue
```

Normalize all transition probabilities

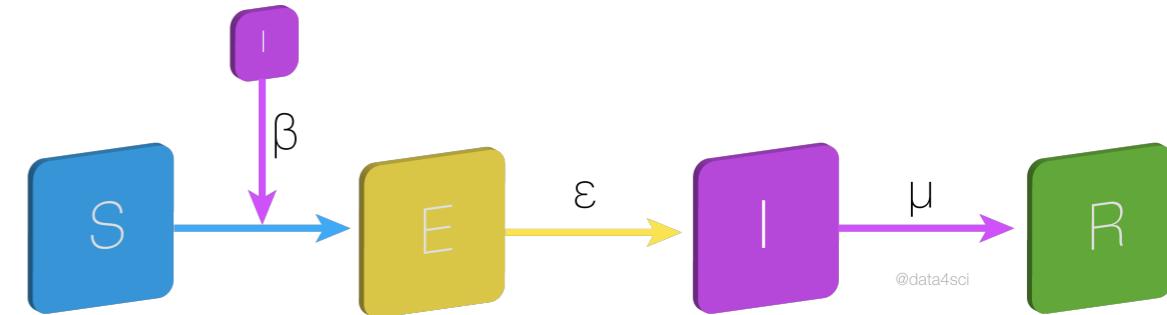
Choose one of the possibilities at random

If a transition occurred, update the node state

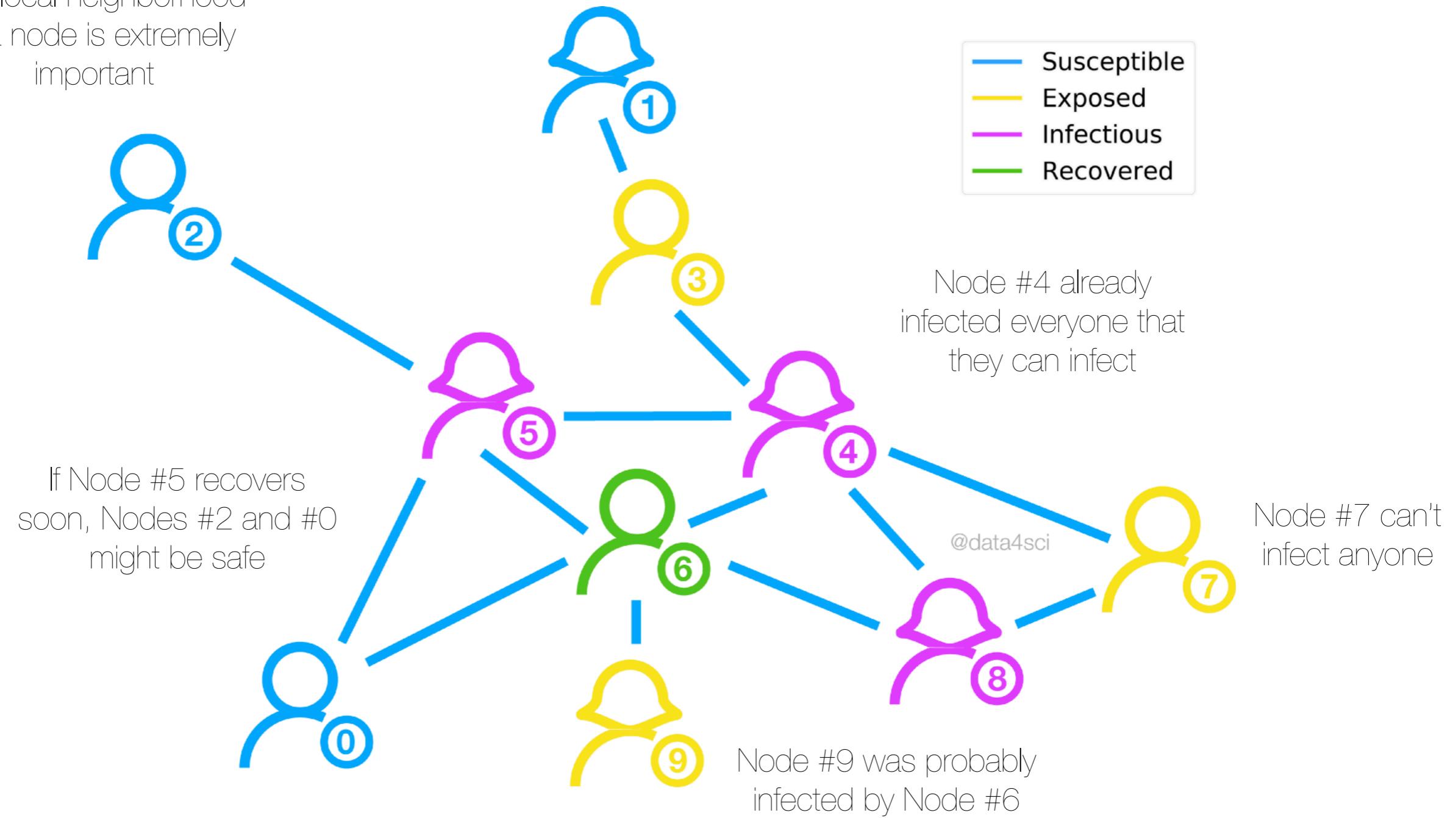
NetworkEpiModel.py

@bgoncalves

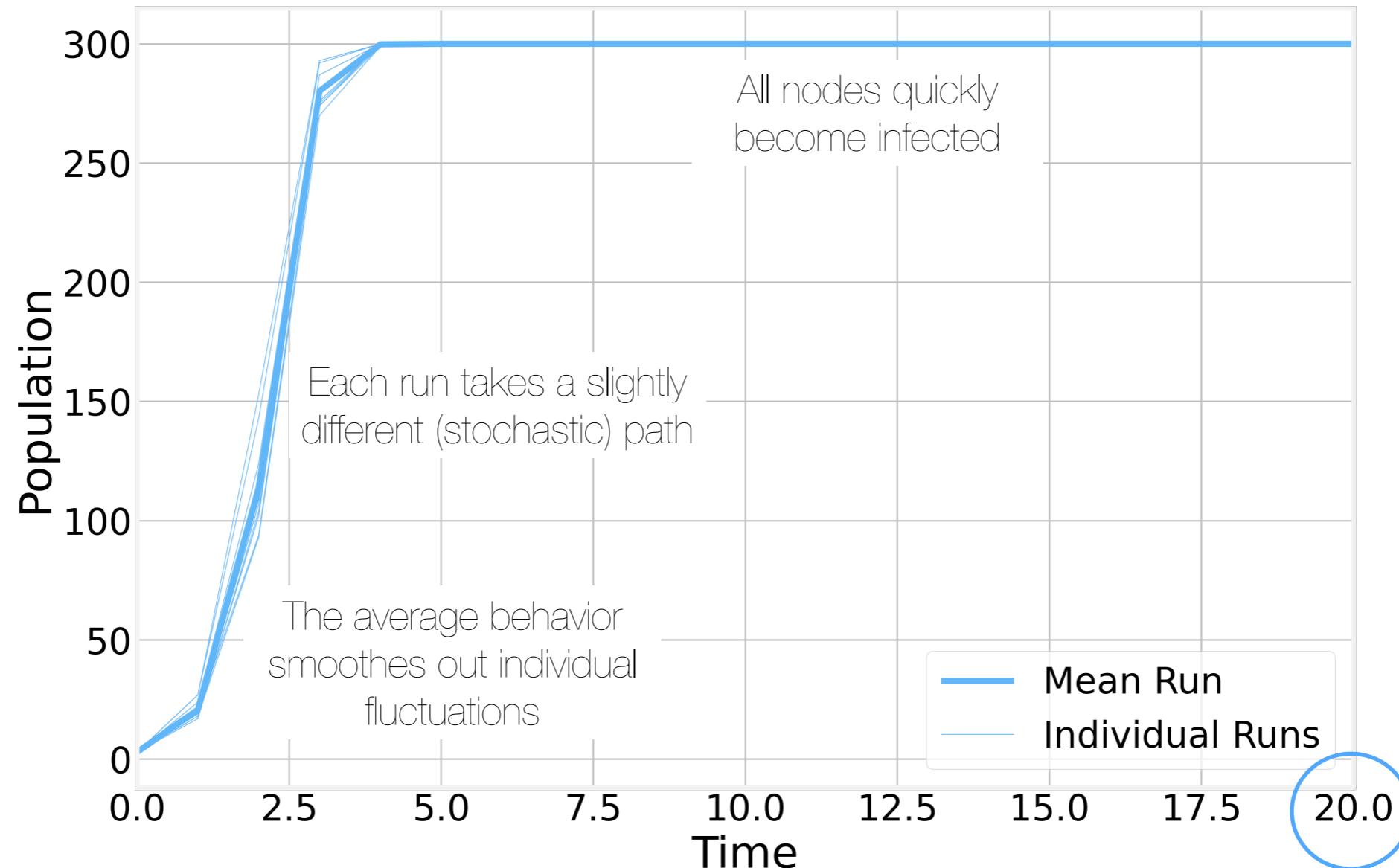
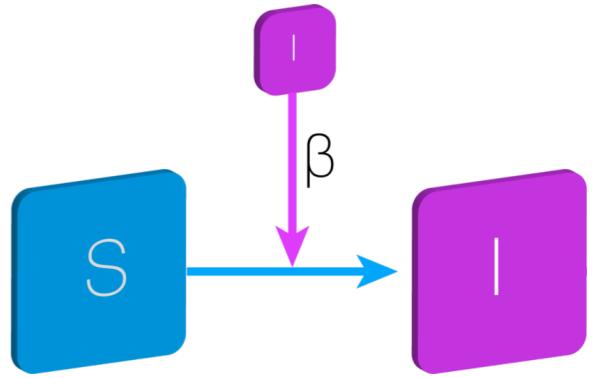
Network Effects



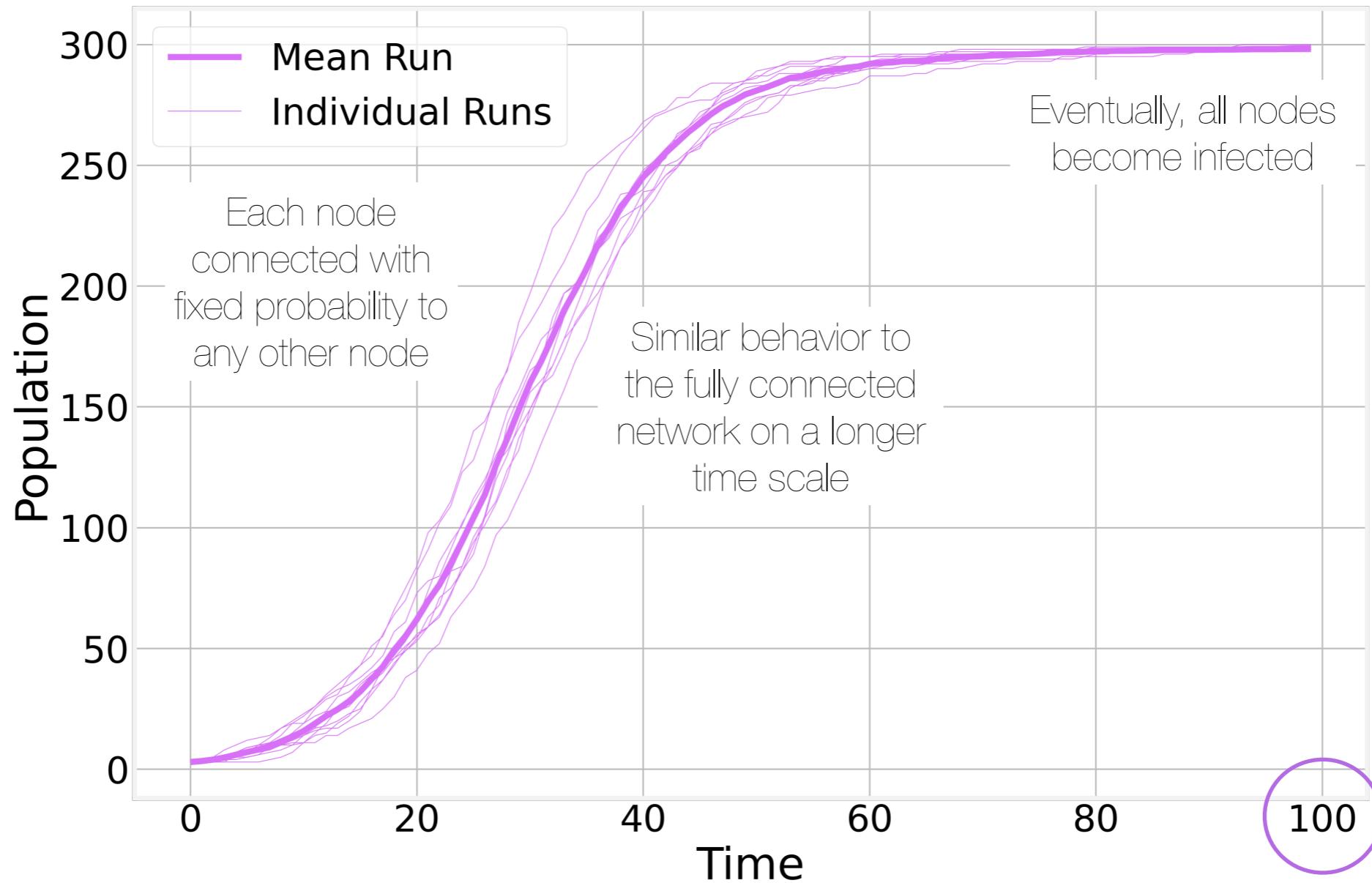
The local neighborhood
of a node is extremely
important



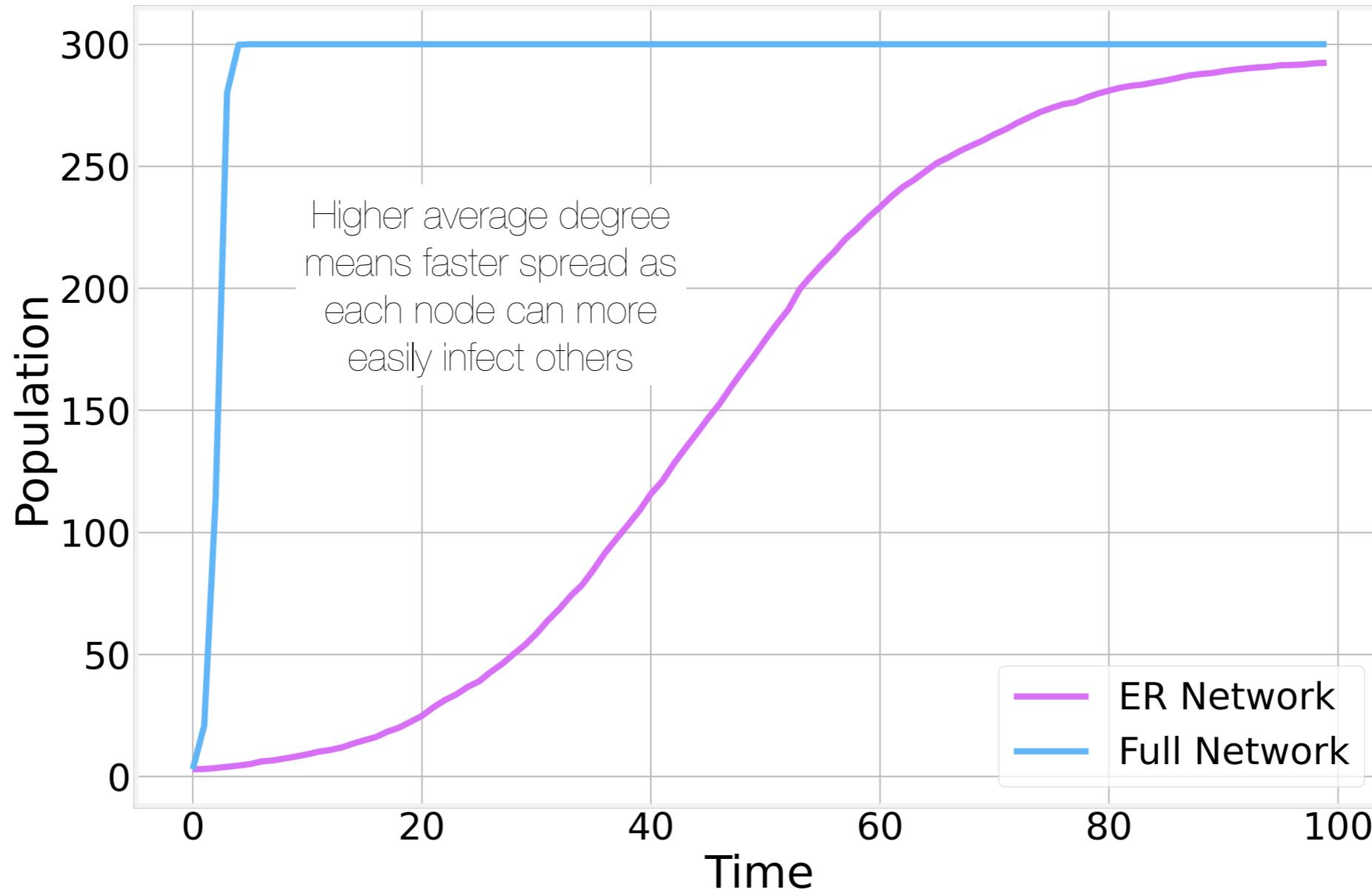
Fully Connected Network



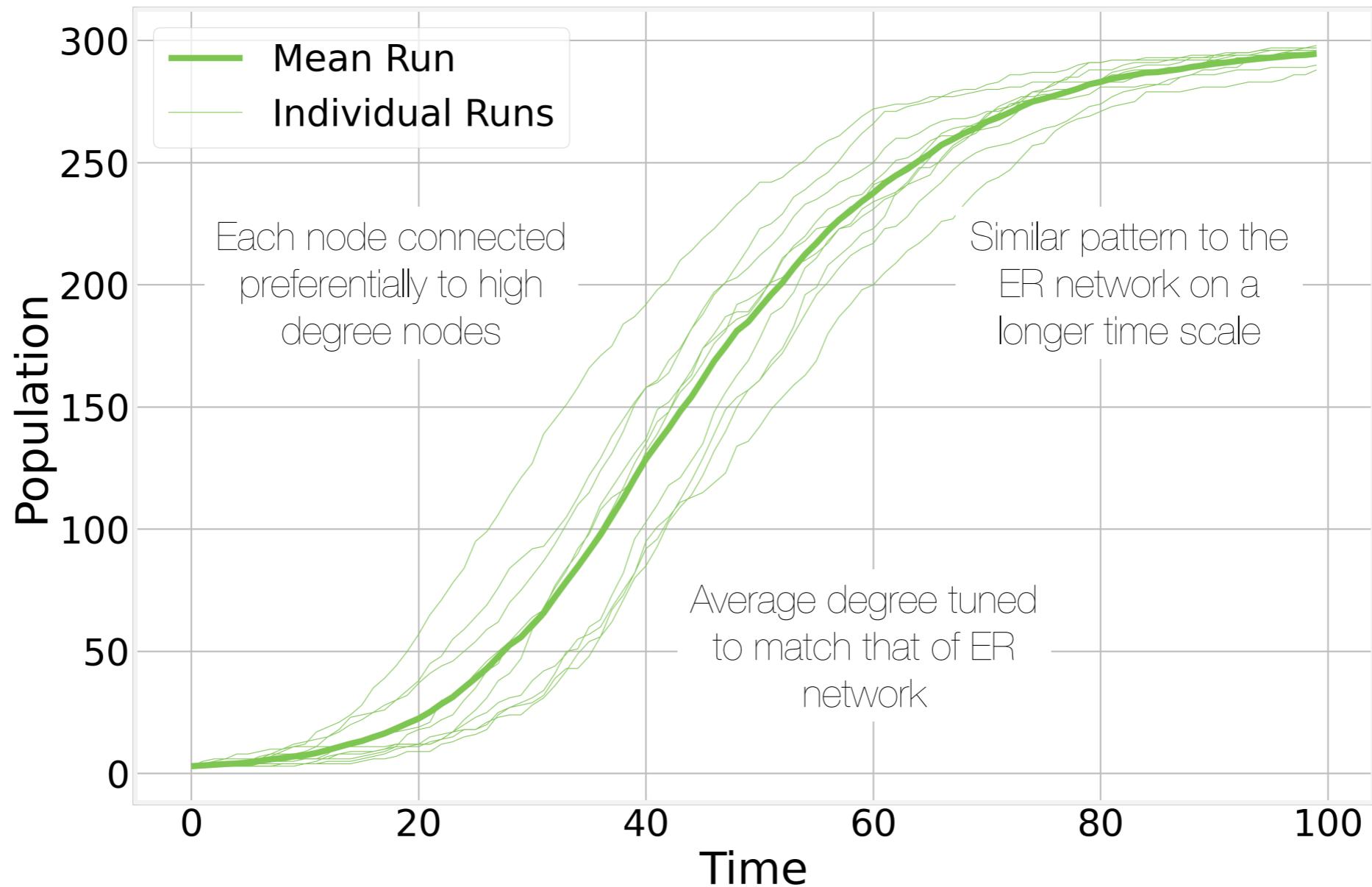
ER Network



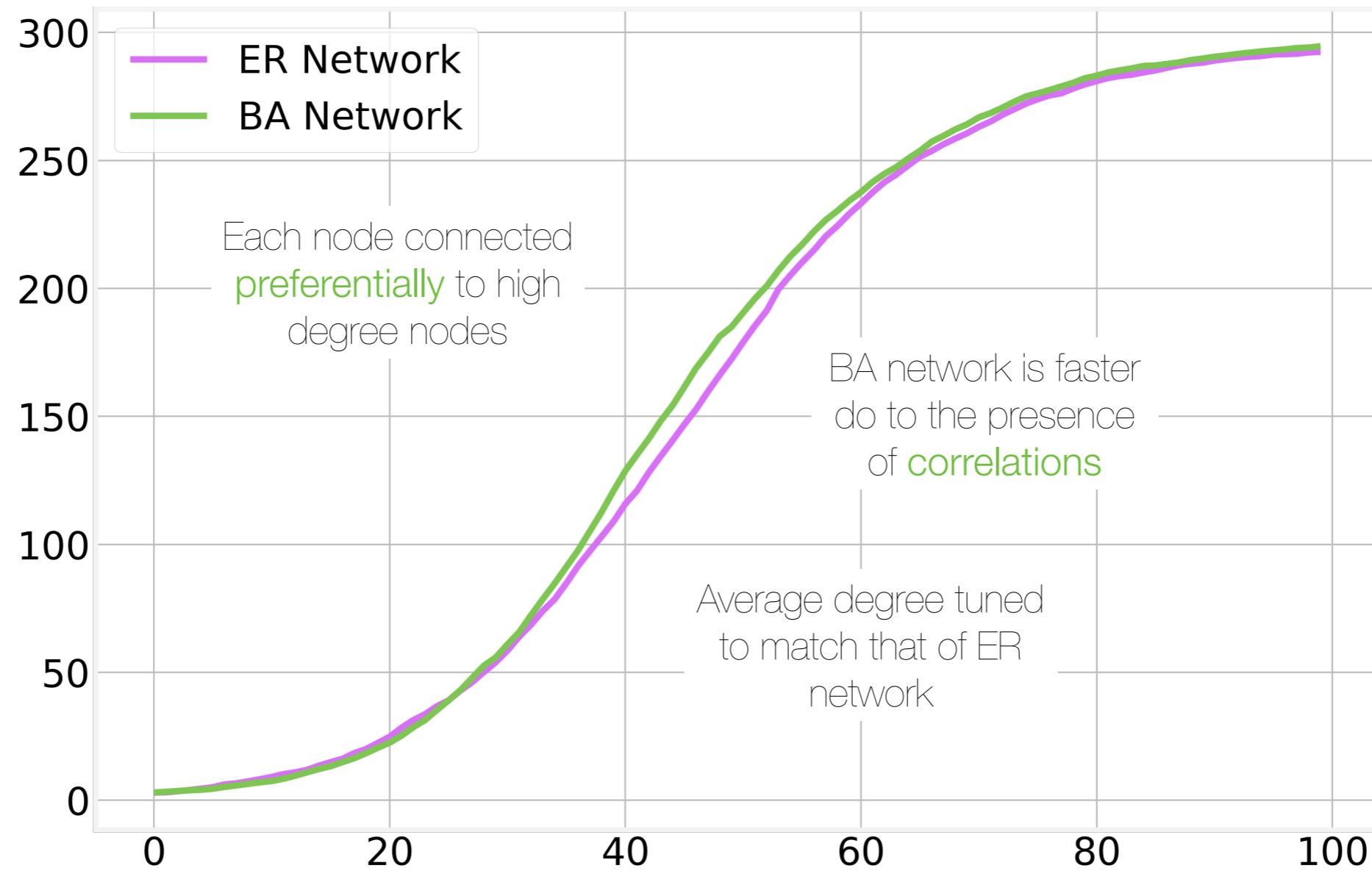
Degree Comparison



BA Network

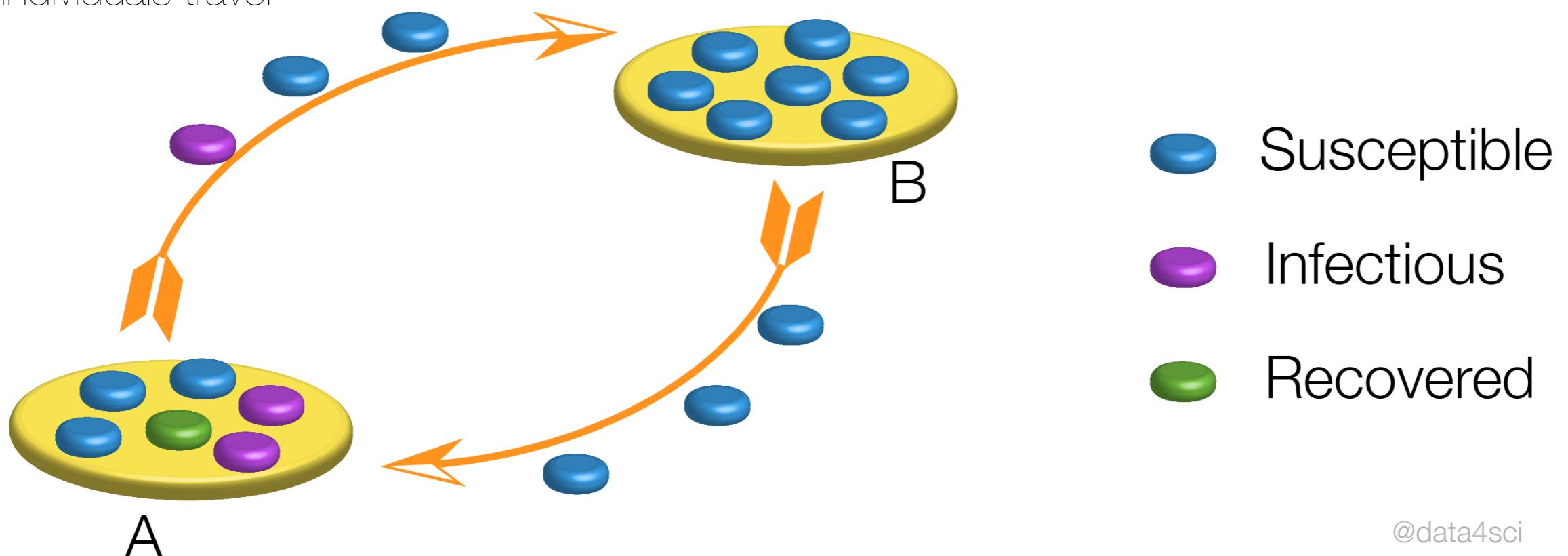


Topology Comparison



Metapopulation Models

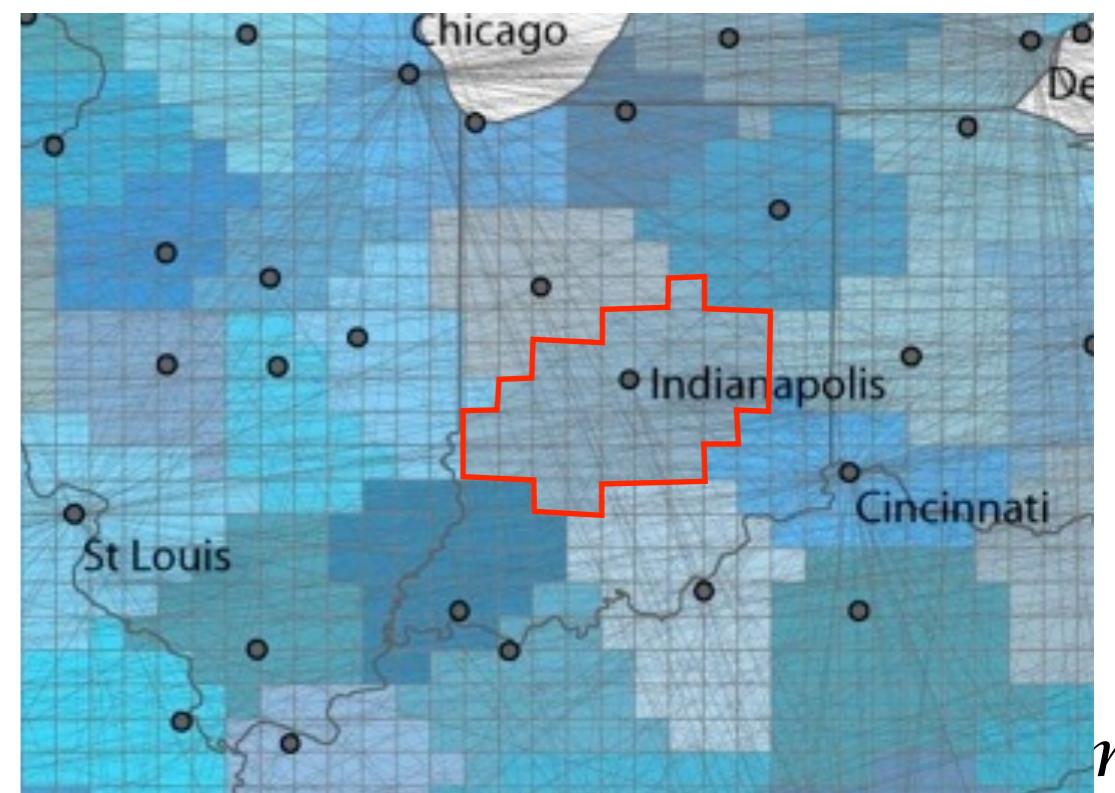
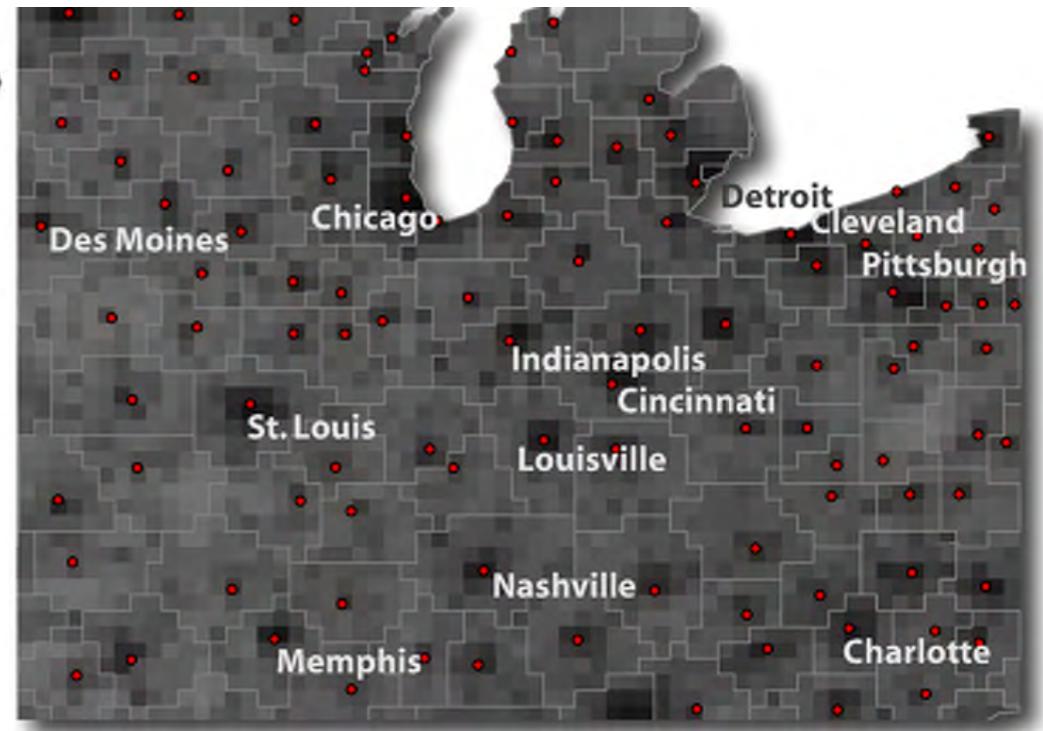
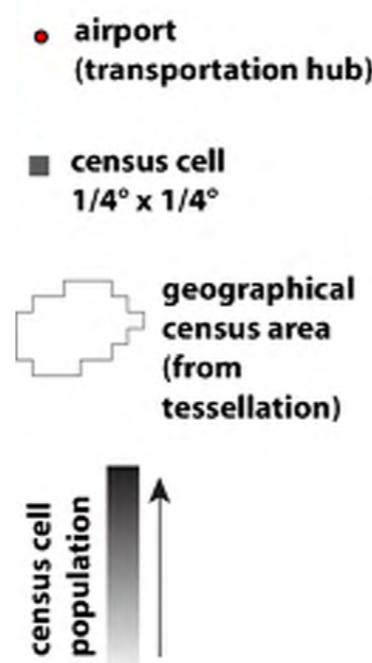
- Metapopulation models divide the full population into inter-connected sub-populations.
- Each sub-population can be considered as well mixed and homogeneous, corresponding to a city or neighborhood
- Connections between subpopulations represent roads, flights, trains, etc
- An infection starting in one sub-population will eventually spread to other populations as individuals travel



@data4sci

Population Distribution

PNAS 106, 21484 (2009)

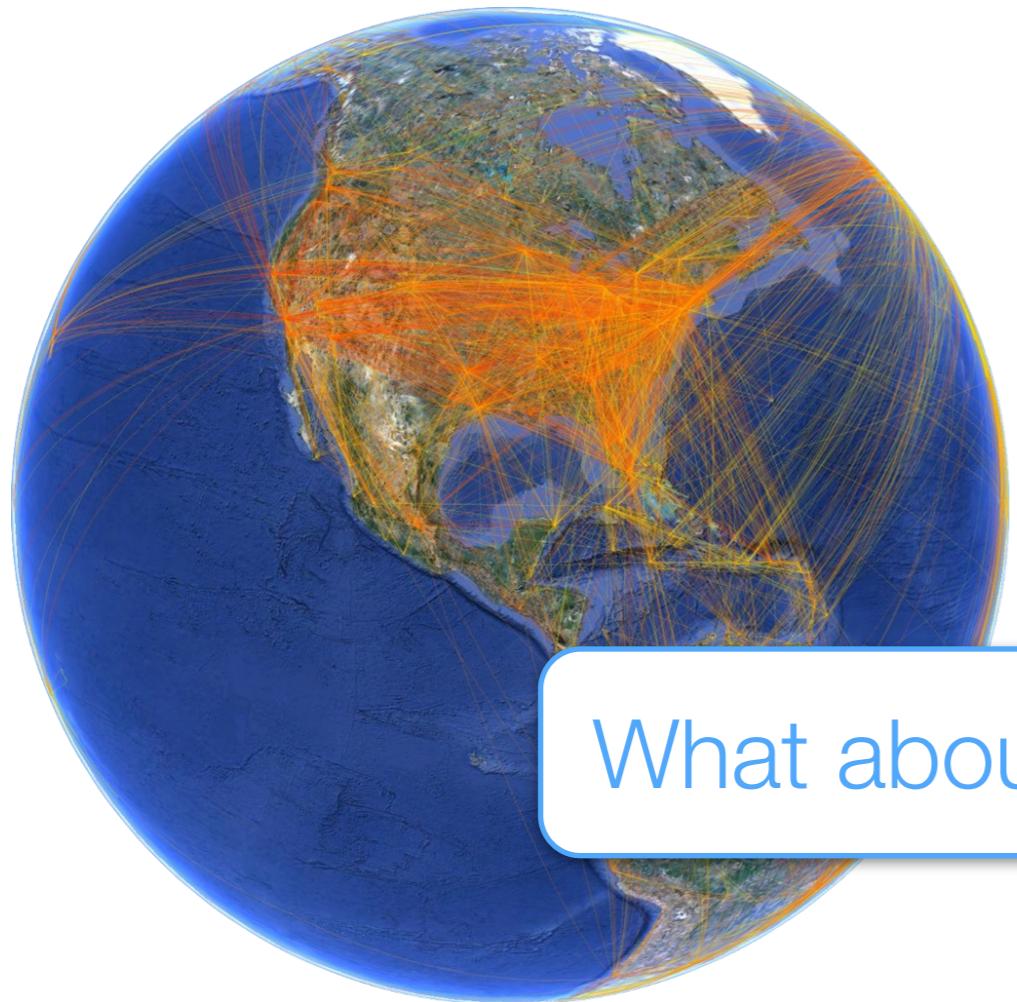


Complete IATA and OAG databases:

3362 airports worldwide
220 countries

Long Range Mobility

PNAS 106, 21484 (2009)



$$P_{jl} = \frac{w_{jl}}{N_j} \Delta t$$

- Probability that any individual in class X travel from $j \rightarrow l$

What about Short Range Mobility?

w
population

Complete IATA and OAG databases:

3362 airports worldwide

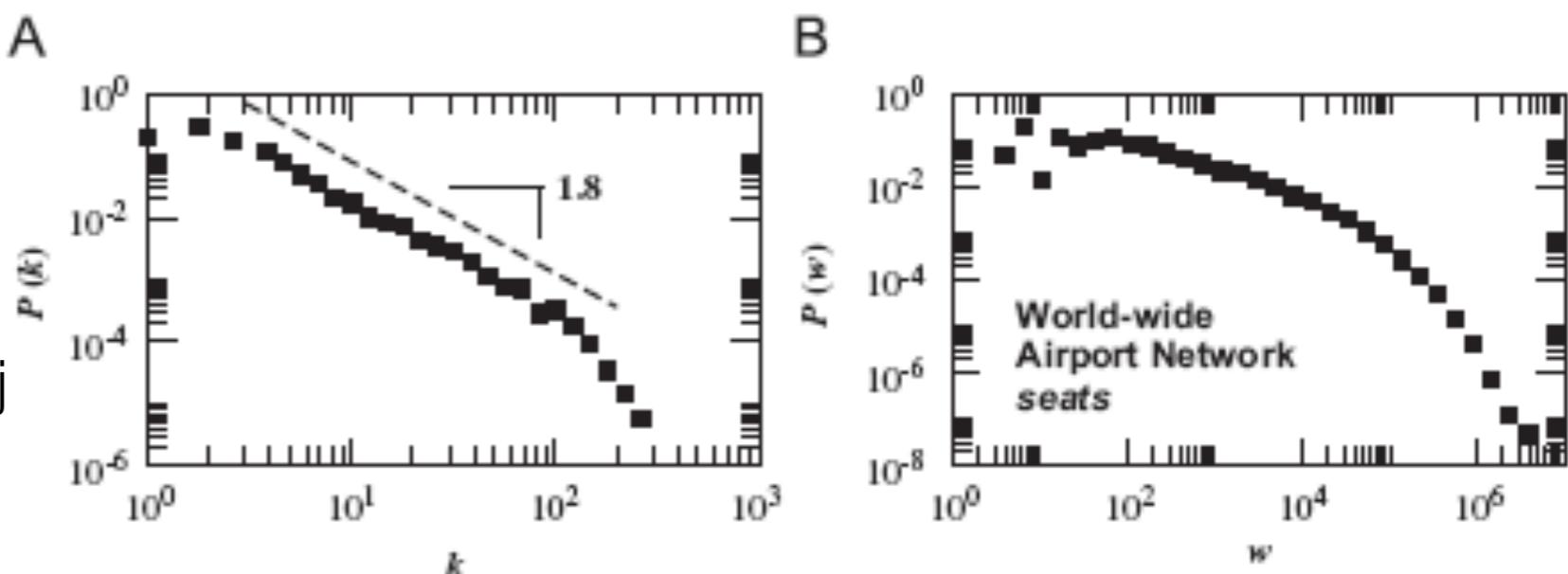
220 countries

> 20,000 connections

w_{ij} #passengers on connection i-j

>99% total traffic

@bgoncalves



Short Range Mobility

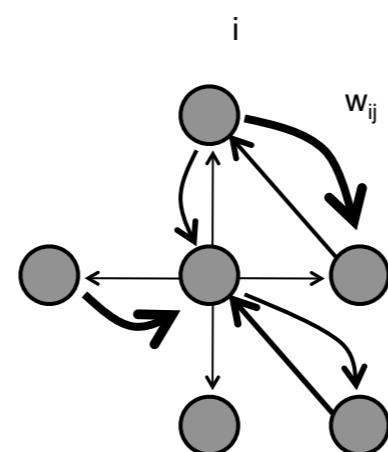
PNAS 106, 21484 (2009)



Table 1

Commuting networks in each continent. Number of countries (N), number of administrative units (V) and inter-links between them (E) are summarized.

Continent	N	V	E
Europe	17	65,880	4,490,650
North America	2	6986	182,255
Latin America	5	4301	102,117
Asia	4	4355	380,385
Oceania	2	746	30,679
Total	30	82,268	5,186,186

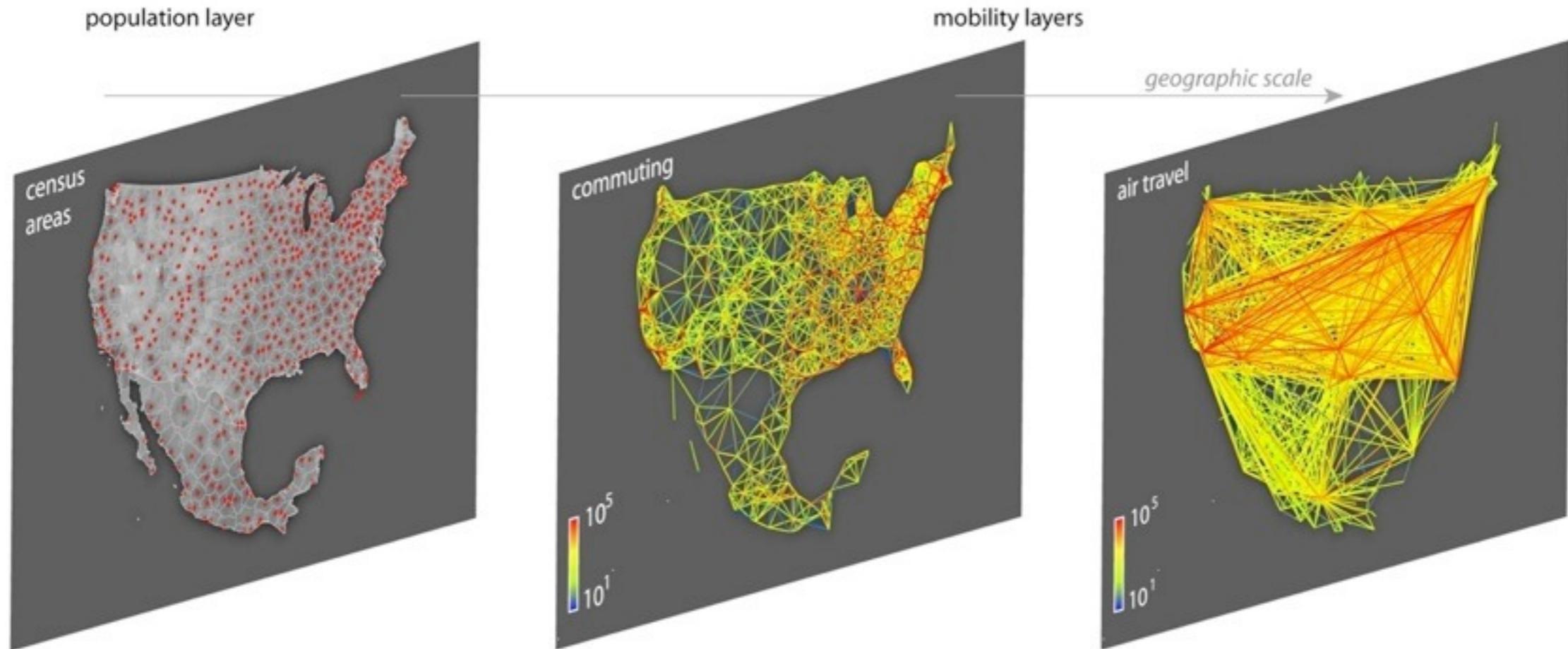


each node i : subpopulation (census area)

each link ($i \rightarrow j$) : interaction between subpopulations i and j

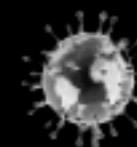
weight w_{ij} : number of people commuting from i to j per unit time

Global Epidemic and Mobility Modeler Platform



Parameter	Value	Description
β	from R_0	transmission probability
ε^{-1}	1.9 [1.1-2.5] d	average latency period
μ^{-1}	3 [3-5] d	average infectious period
p_t	50%	probability of traveling for infectious individuals
p_a	33%	probability of being asymptomatic
r_β	50%	relative infectiousness of asymptomatic infectious individuals

Invasion Tree



GLEaMviz.org



Jun 23, 2009

 MoBS



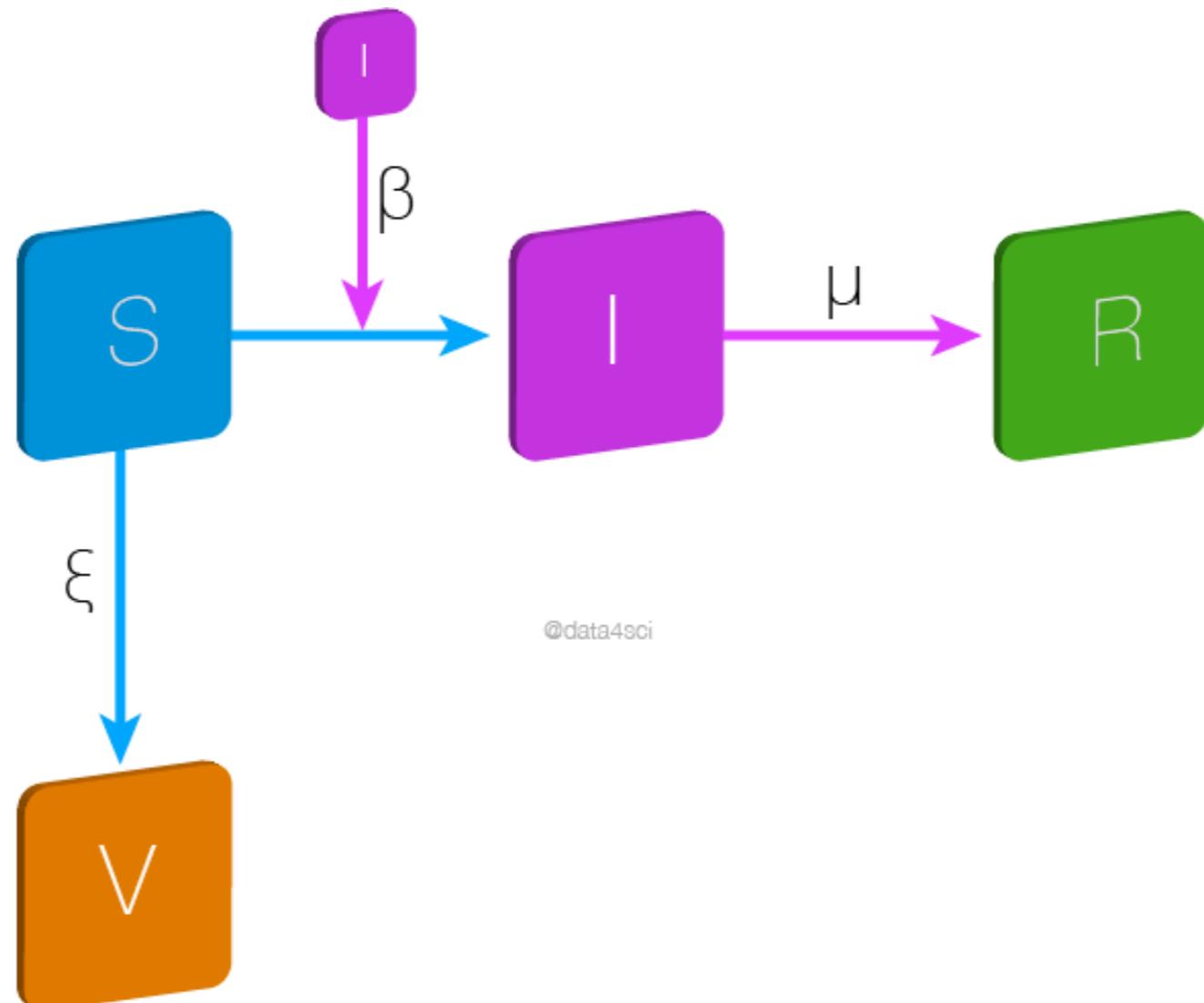
Code - Network Models
<https://github.com/DataForScience/CoVID19>

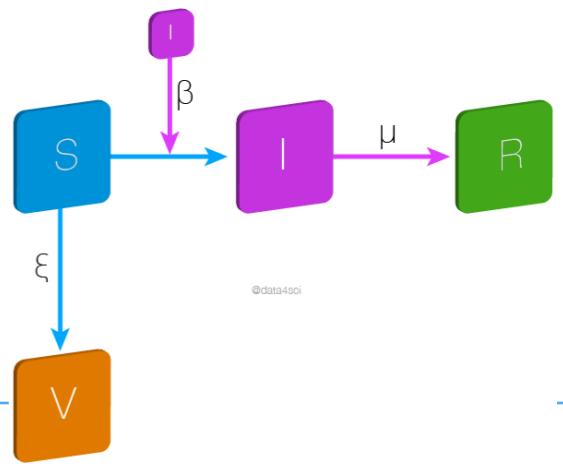


5. Vaccination

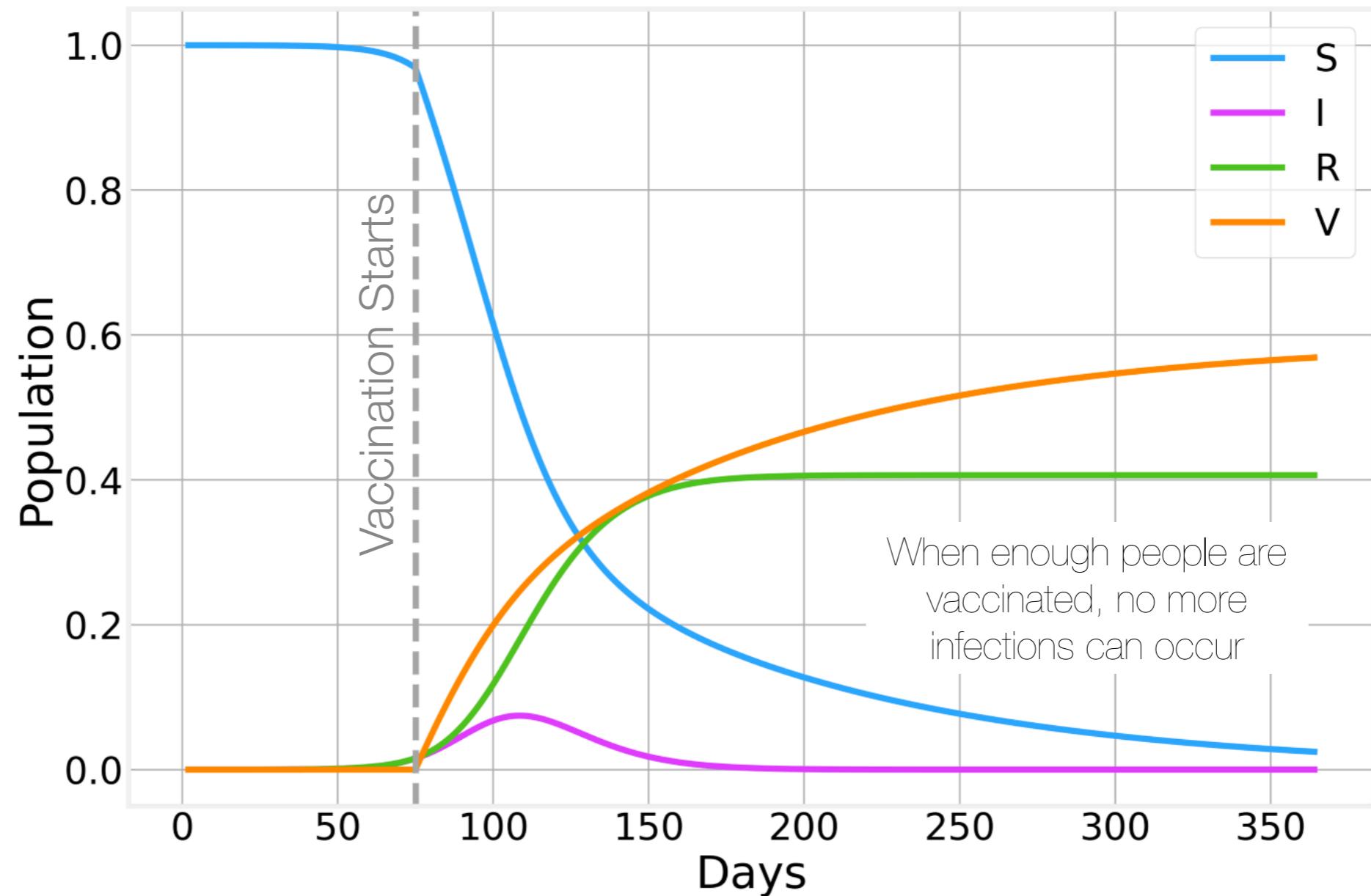
Modeling vaccination

If the vaccine is 100% effective, vaccination effectively removes individuals from the susceptible pool, choking the epidemic

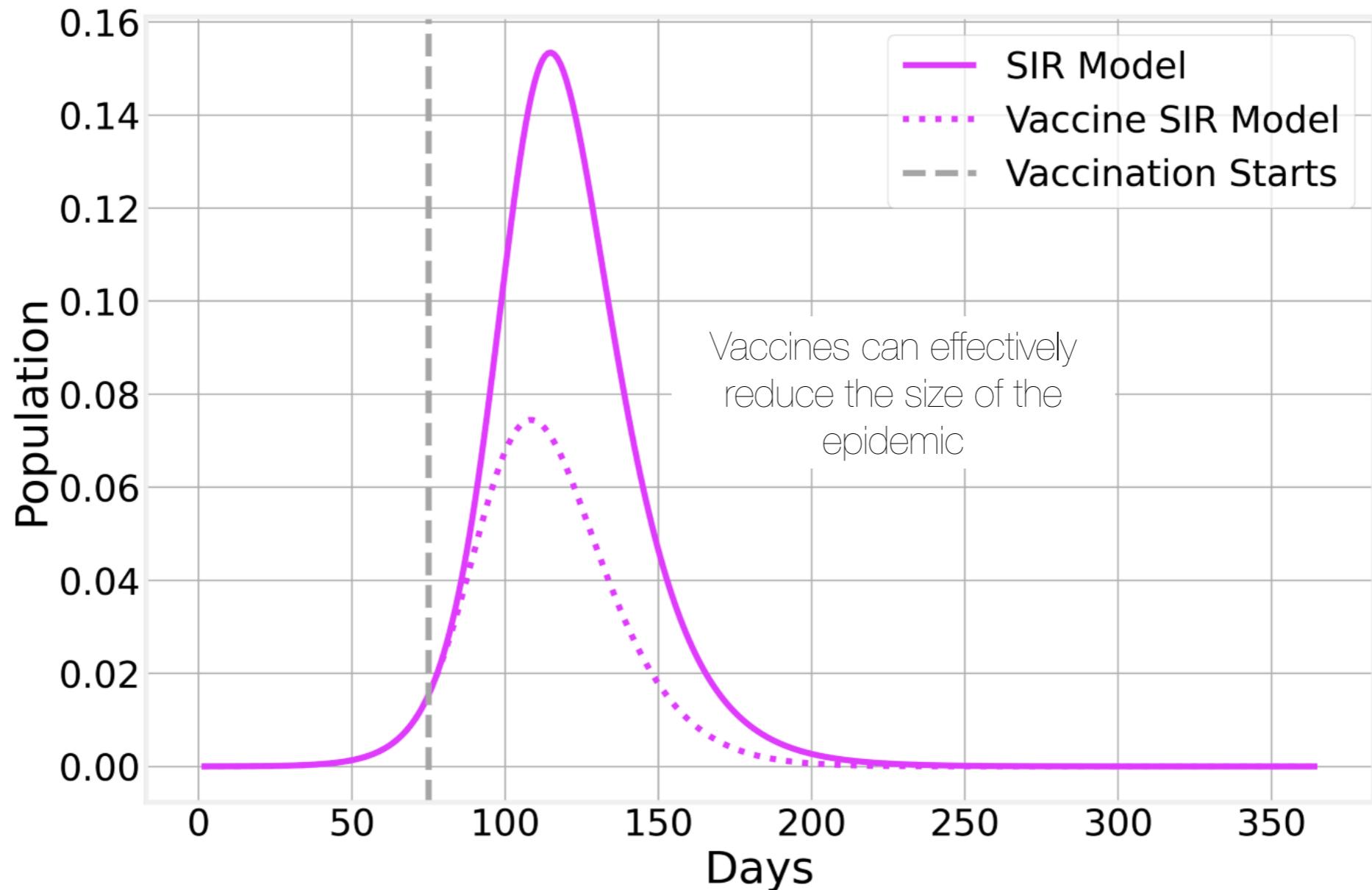




Perfect vaccines



Perfect Vaccines



Herd Immunity

- Herd Immunity is the concept that if a large enough percentage of the population is immune, then the disease can't spread for lack of susceptible individuals to infect
- A rough estimate of herd immunity can be easily derived from the SIR model equations. As we saw above:

$$\frac{dI}{dt} = +\beta S \frac{I}{N} - \mu I$$

- This derivative becomes negative when:

$$0 > +\beta S \frac{I}{N} - \mu I$$

- or, equivalently:

$$N \frac{\mu}{\beta} > S \rightarrow S < \frac{N}{R_0}$$

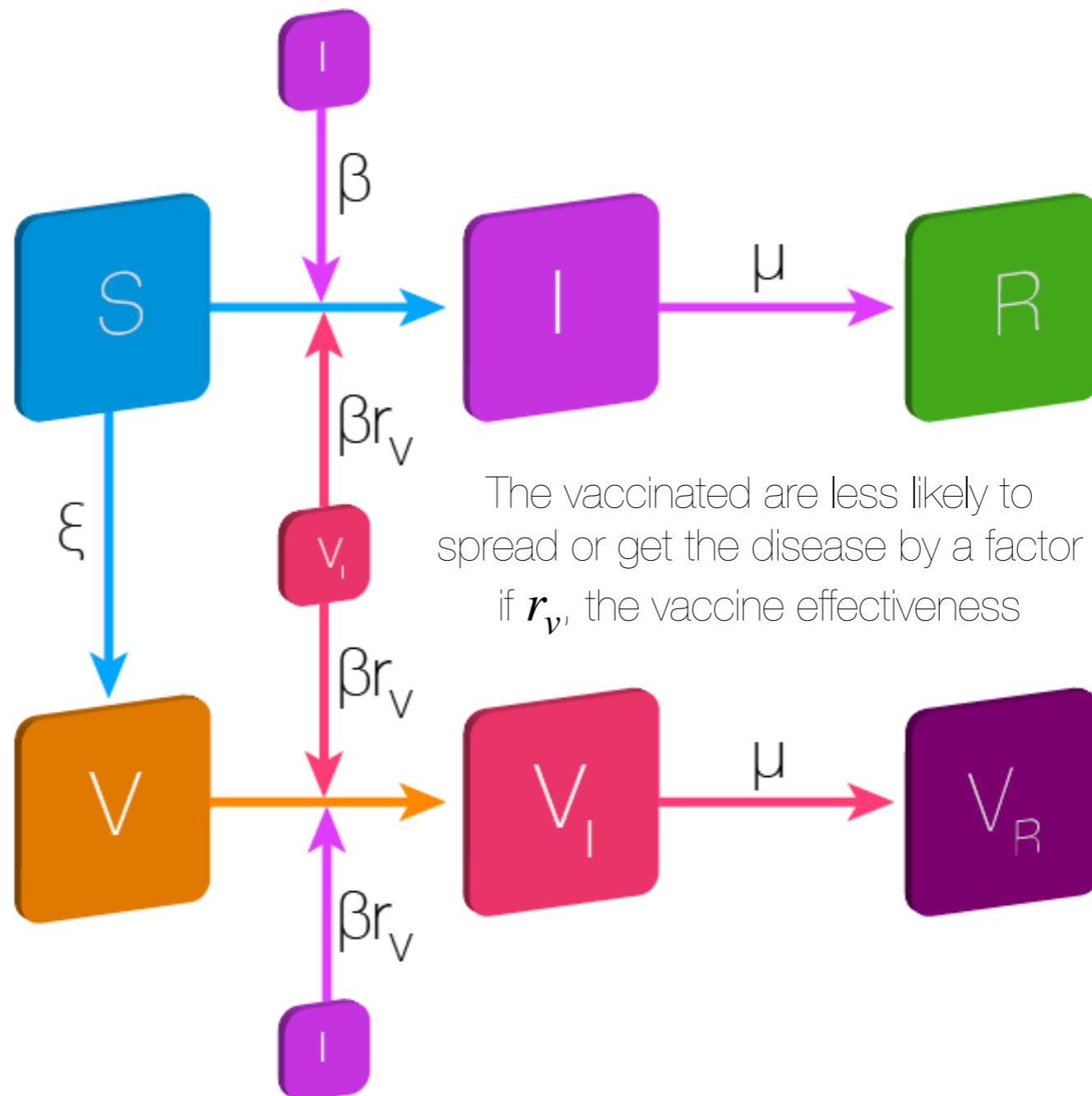
- So the fraction of immune in the population must be larger than:

$$\text{Immune} > 1 - \frac{1}{R_0}$$

- Which gets increasingly larger as R_0 increases

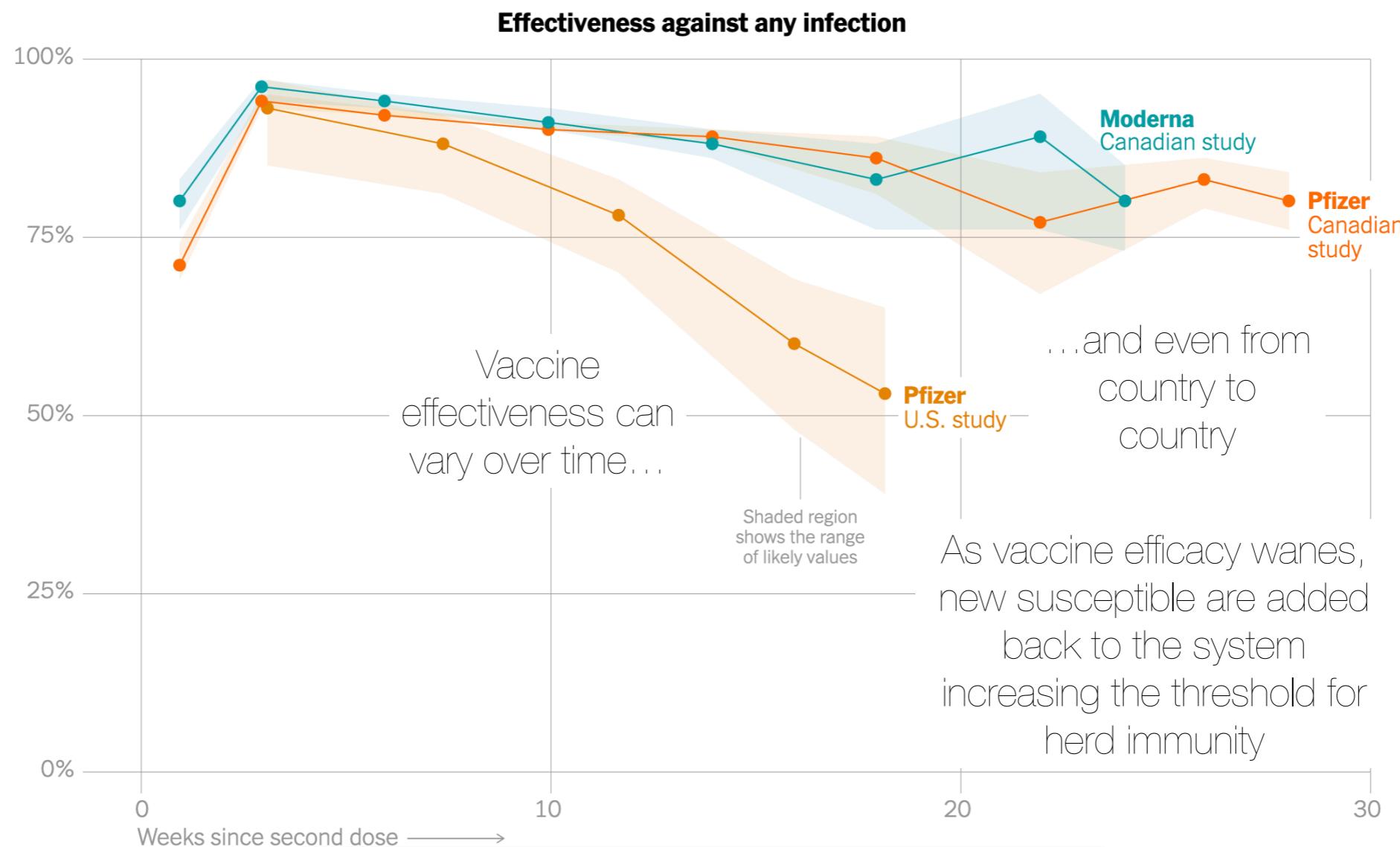
This quantity doesn't depend on whether immunity was obtained through infection, or through vaccination

Modeling vaccination

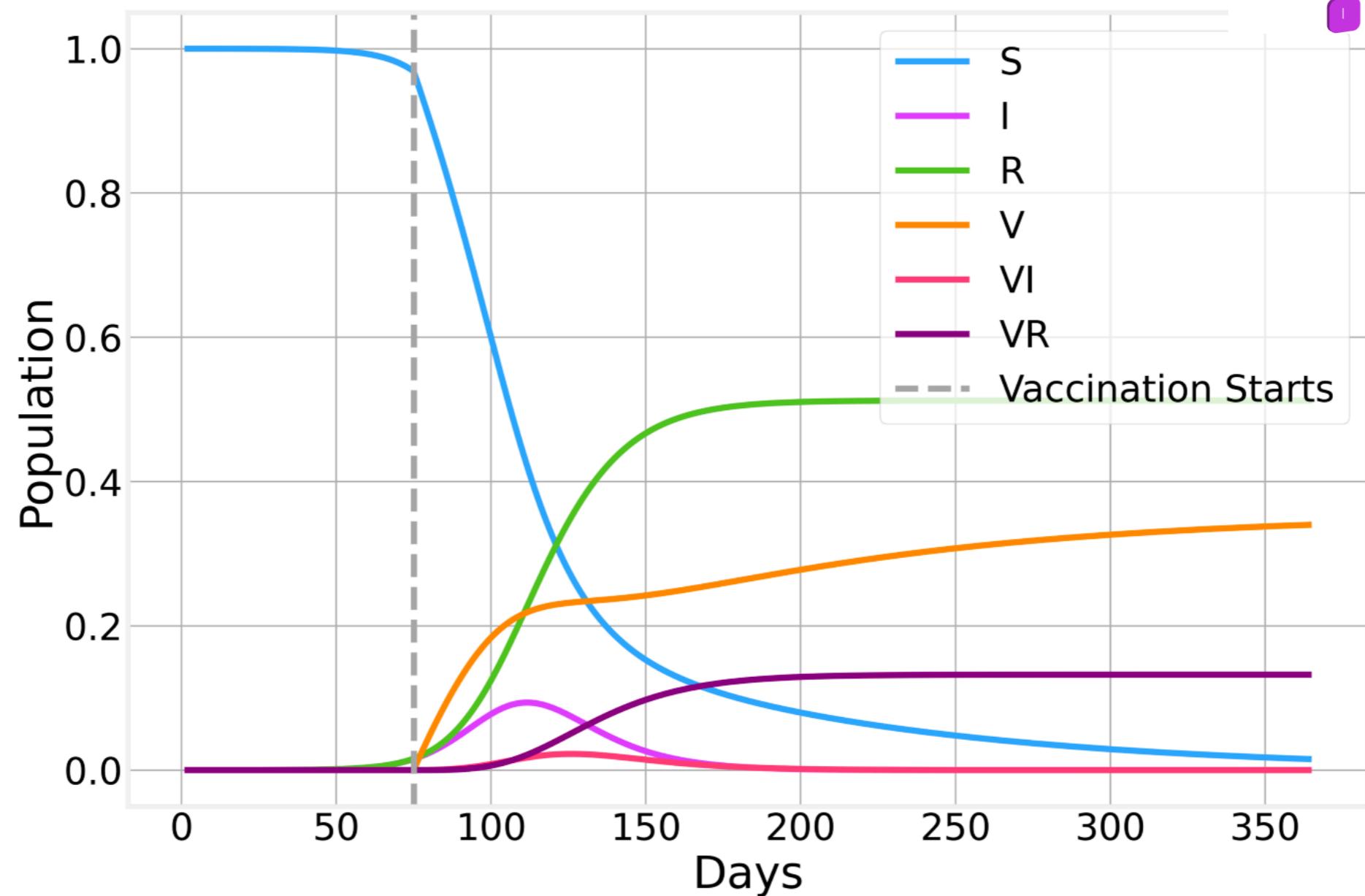
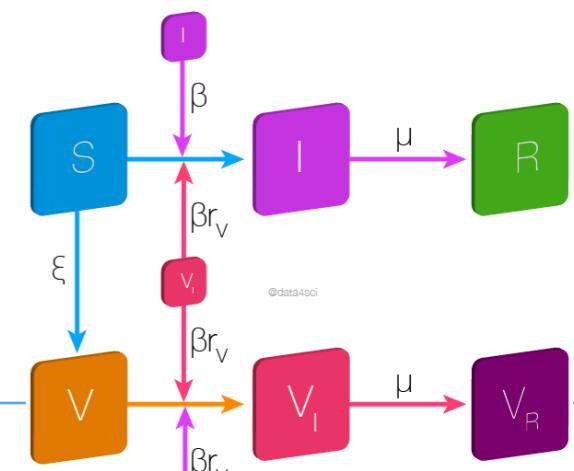


Modeling Vaccination

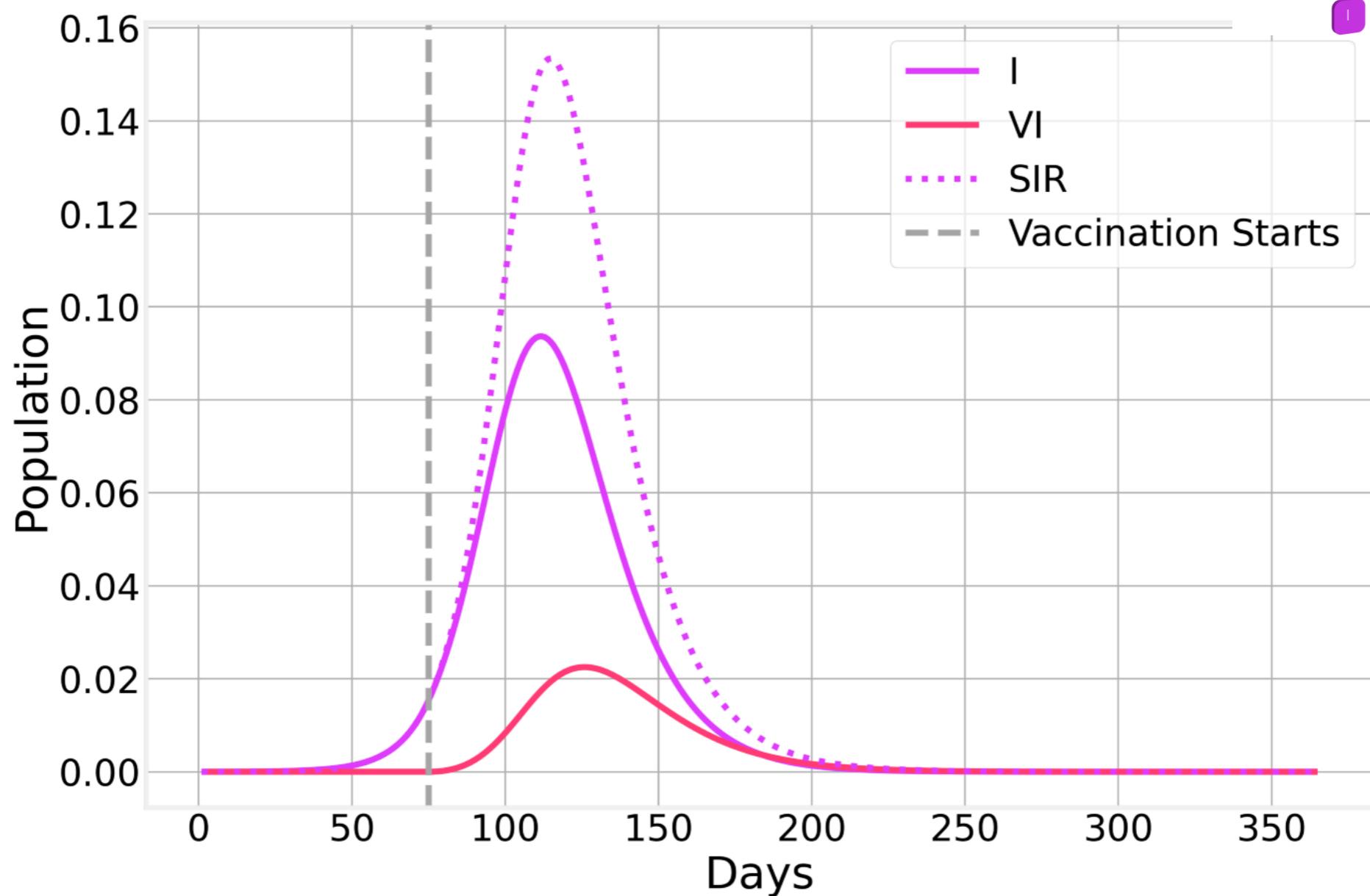
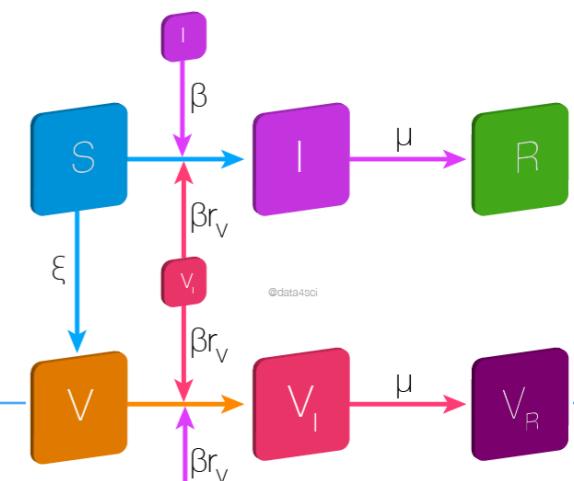
<https://www.nytimes.com/interactive/2021/11/11/science/vaccine-waning-immunity.html>



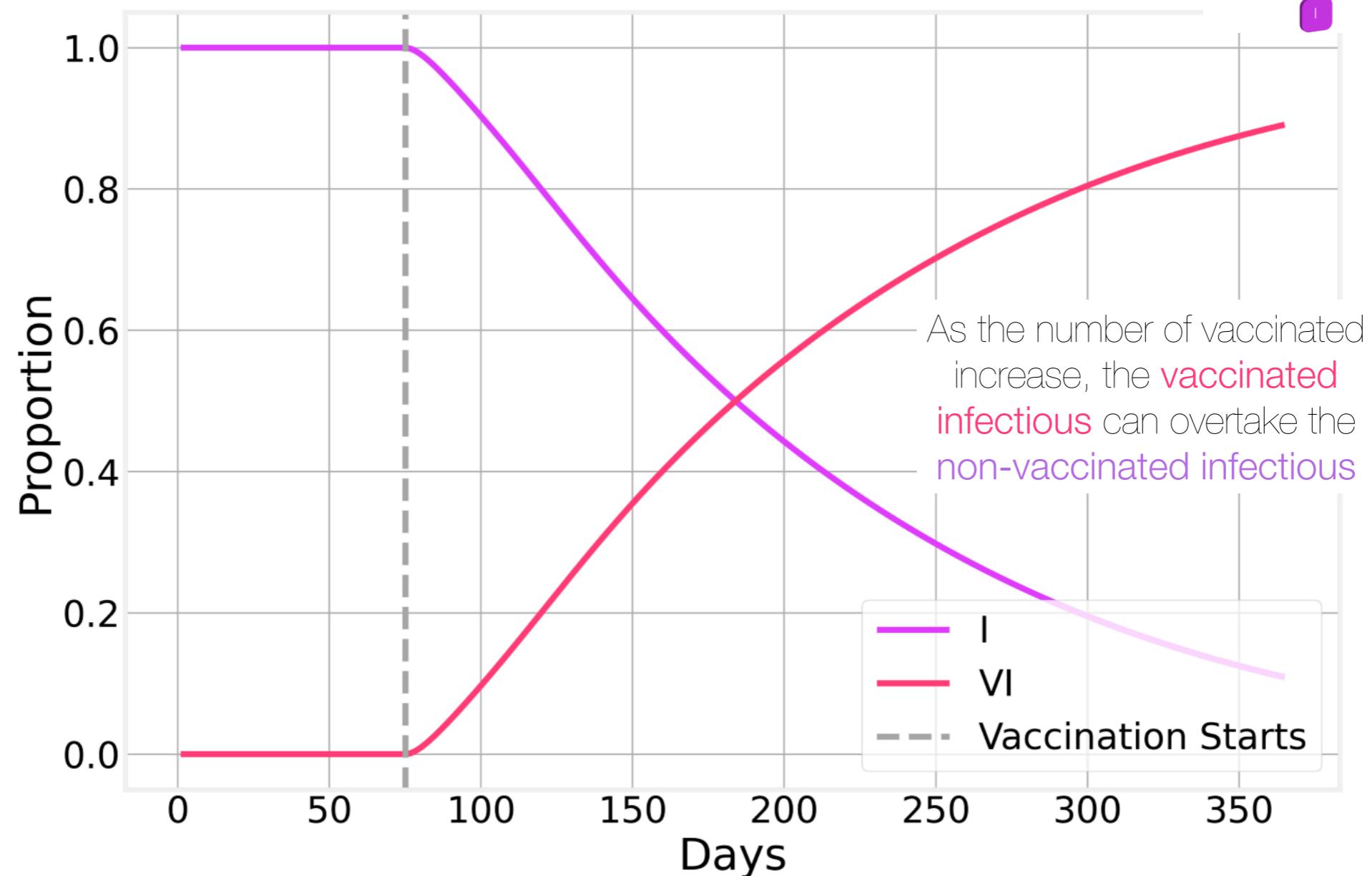
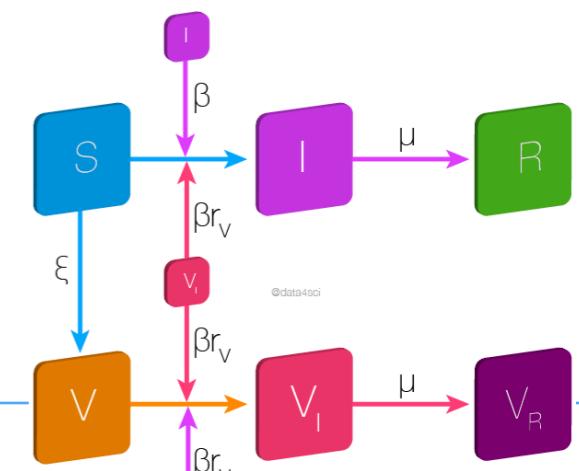
Real-World Vaccines



Real-World Vaccines



Real-World Vaccines

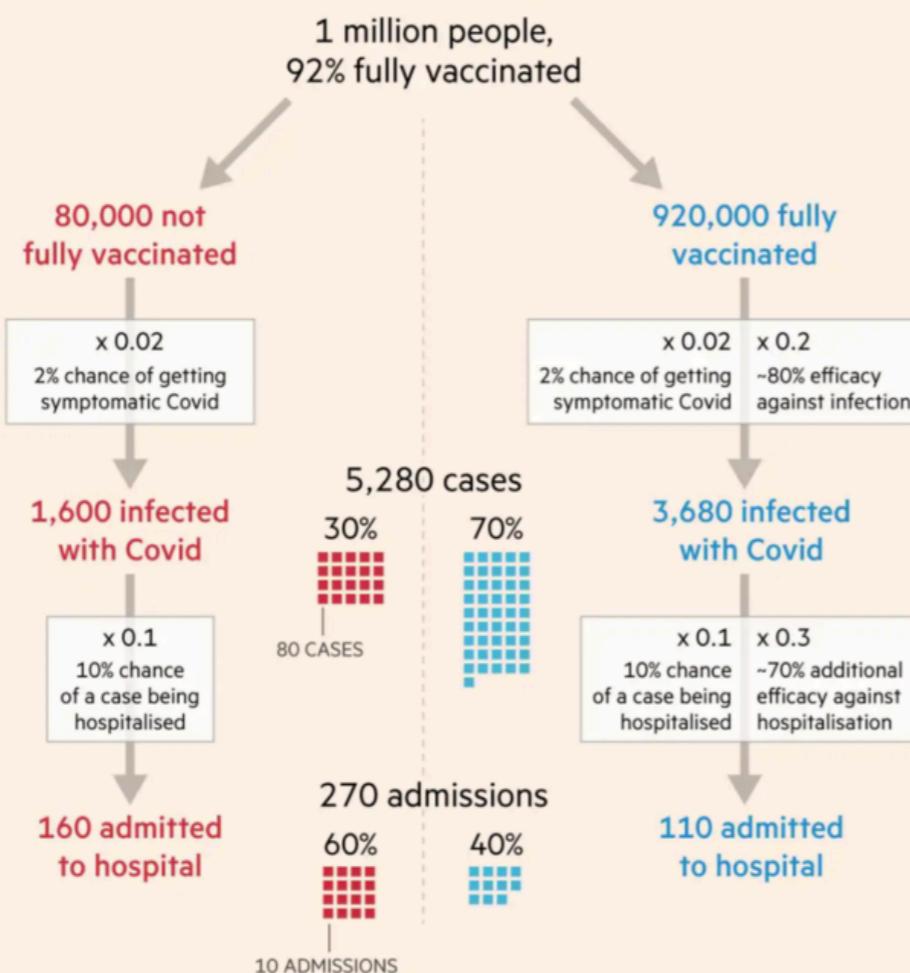


Real-World Vaccines

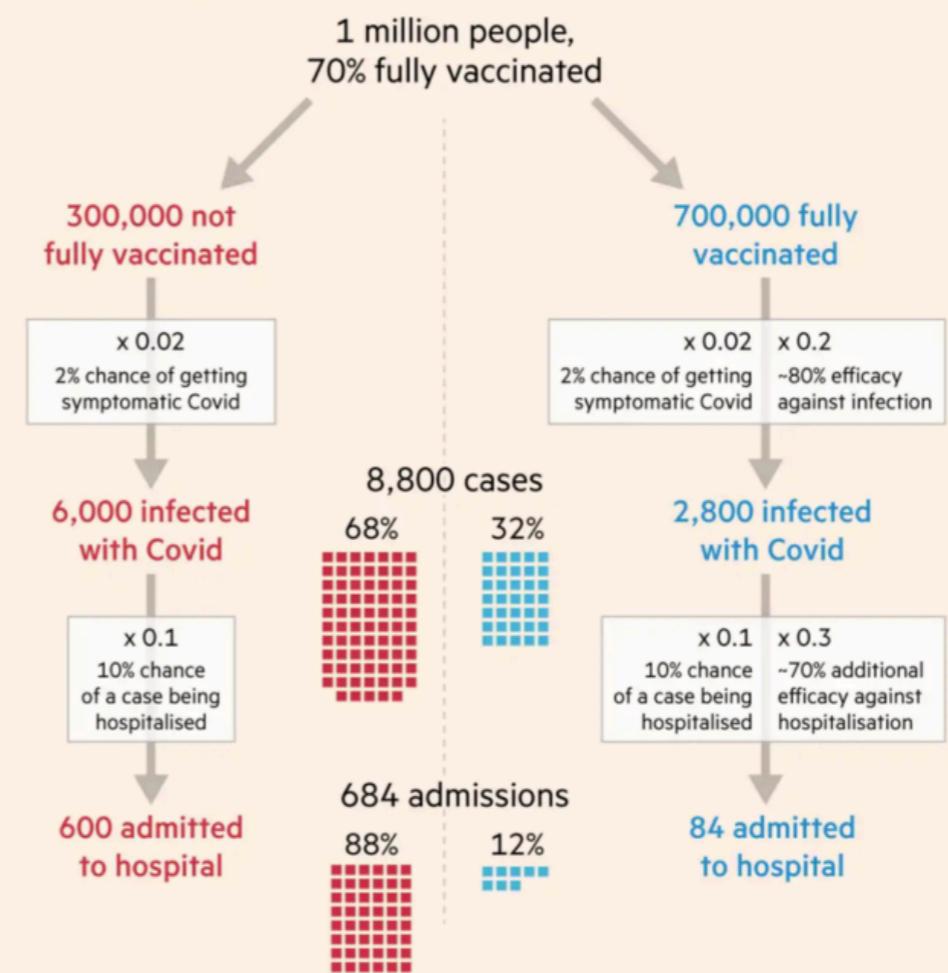
<https://www.ft.com/content/0f11b219-0f1b-420e-8188-6651d1e749ff>

Why a high proportion of double-jabbed people among hospitalisations does not mean the vaccines are failing

When a **very high share of people are fully vaccinated**, a large share of cases and hospitalisations will be double-jabbed people despite the vaccines working very well



When a **lower share are fully vaccinated**, the double-jabbed are a smaller share of cases and hospitalisations, but far more people end up in hospital overall



Sources: Left-hand scenario based on vaccination rates for people aged 50+ in England, vaccine efficacy from PHE. Right-hand scenario is hypothetical
© FT

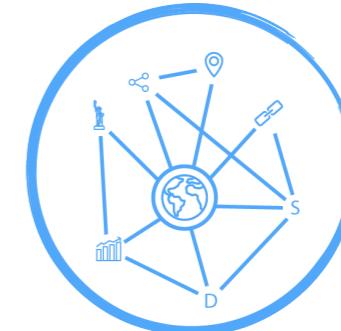


Code - Vaccination
<https://github.com/DataForScience/CoVID19>

Discussion

- Epidemic processes are fascinating and extremely complex
- CoVID-19 has resulted in increased interest in Epidemic modeling, while generating huge amounts of data
- Data Science has a fundamental role to play in assessing, forecasting and controlling this (and future) pandemics
- Networks play a fundamental role at every stage and level of detail:
 - DAGs describe the transitions between compartments
 - Social networks describe who interacts with whom
 - Transportation networks impact travel and social activity
- Careful model fitting and training can provide unique insights into an ongoing pandemic, helping guide public health measures

Thank you!



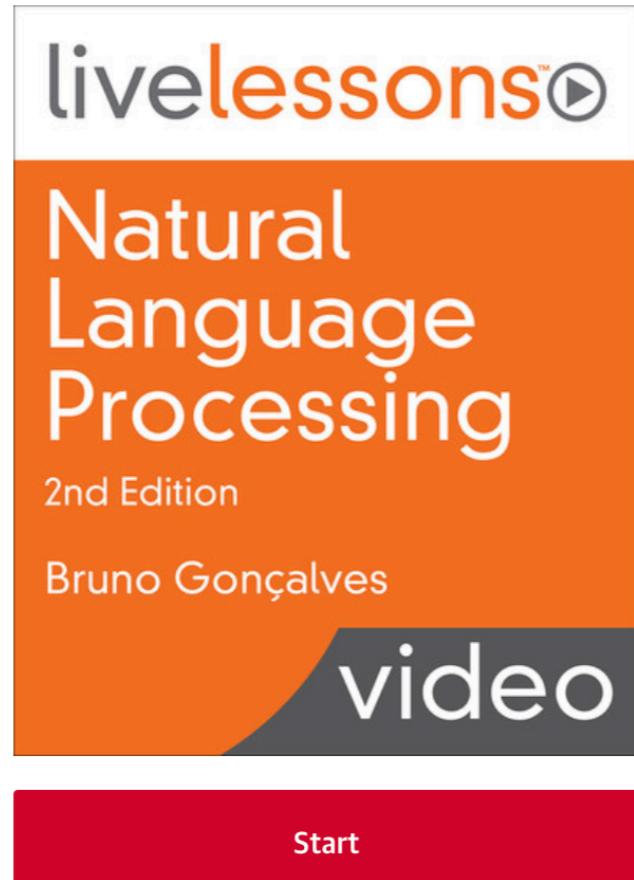
graphs4sci.substack.com

- Hope you enjoyed this tutorial. Looking forward to your questions and thoughts.
- You can subscribe to the [graphs4sci substack](#) to receive Graph related content in your inbox every couple of weeks
- Don't forget to download the Jupyter notebooks and the slide deck from [GitHub](#):
<https://github.com/DataForScience/CoVID19>
- Please feel free to reach out on Social Media!
- [Website](#): www.data4sci.com
- [Twitter](#): @data4sci
- [Email](#): bgoncalves@data4sci.com

Natural Language Processing, 2nd Edition

Write the [first review](#)

By [Bruno Gonçalves](#)



TIME TO COMPLETE:

5h 23m

TOPICS:

[Natural Language Processing](#)

PUBLISHED BY:

[Addison-Wesley Professional](#)

PUBLICATION DATE:

October 2021

https://bit.ly/NLP_LL

5 Hours of Video Instruction

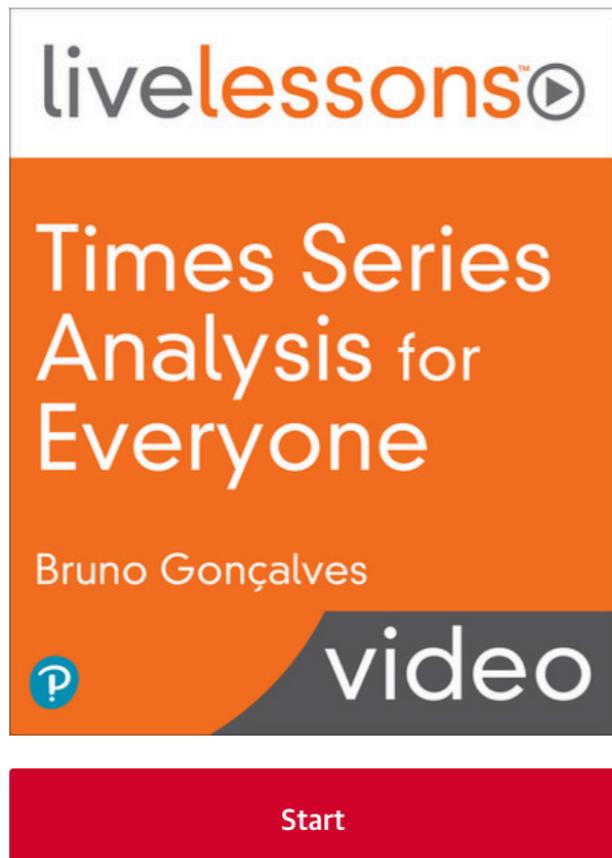
Overview

Natural Language Processing LiveLessons covers the fundamentals of Natural Language Processing in a simple and intuitive way, empowering you to add NLP to your toolkit. Using the powerful NLTK package, it gradually moves from the basics of text representation, cleaning, topic detection, regular expressions, and sentiment analysis before moving on to the Keras deep learning framework to explore more advanced topics such as text classification and sequence-to-sequence models. After successfully completing these lessons you'll be equipped with a fundamental and practical understanding of state-of-the-art Natural Language Processing tools and algorithms.

Times Series Analysis for Everyone

★★★★★ [1 review](#)

By [Bruno Gonçalves](#)



TIME TO COMPLETE:

6h

TOPICS:

[Time Series](#)

PUBLISHED BY:

[Pearson](#)

PUBLICATION DATE:

November 2021

https://bit.ly/Timeseries_LL

6 Hours of Video Instruction

The perfect introduction to time-based analytics

Overview

Times Series Analysis for Everyone LiveLessons covers the fundamental tools and techniques for the analysis of time series data. These lessons introduce you to the basic concepts, ideas, and algorithms necessary to develop your own time series applications in a step-by-step, intuitive fashion. The lessons follow a gradual progression, from the more specific to the more abstract, taking you from the very basics to some of the most recent and sophisticated algorithms by leveraging the statsmodels, arch, and Keras state-of-the-art models.