HOCHSCHULE
MITTWEIDA
UNIVERSITY OF
APPLIED SCIENCES

SIMULATION AND VISUALISATION

Group Members
Santhosh Naredla, Jawad Niazi , Tek Bahadur Bista
50896, 47310, 51042

Predicting the number of reported and unreported cases for the Covid19 epidemic in Germany , Italy, South Korea and UK

SIRU Model

2020

Faculty of **Applied Computer and Bio Sciences**

SIMULATION AND VISUALISATION

Predicting the number of reported and unreported cases for the Covid19 epidemic in Germany , Italy, South Korea and UK

SIRU Model

Author:
Santhosh Naredla, Jawad Niazi , Tek Bahadur Bista
50896, 47310, 51042

Study Programme:
Applied Mathematics for Network and Data Sciences

Seminar Group:
MA18W1-M, MA19W1-M

First Referee:
Kristan Alexander Schneider

Second Referee:
Gideon Akumah Ngwa

Mittweida, August 2020

Abstract

Background

The novel coronavirus (SARS-CoV-2) is currently causing concern in the medical, epidemiological and mathematical communities as the virus is rapidly spreading around the world. Internationally, there are more than 1 200 000 cases detected and confirmed in the world on April 6. The asymptomatic and mild symptomatic cases are just going to be really crucial for us to understand what is driving this epidemic to transmit rapidly. Combining a mathematical model of severe (SARS-CoV-2) transmission with data from Germany, Italy, South Korea and United Kingdom, we provide the epidemic predictions of the number of reported and unreported cases for the SARS-CoV-2 epidemics and evaluate the effectiveness of control measures for each country.

Methods

Methods We combined a mathematical model with data on cumulative confirmed cases from Germany, Italy, South Korea and United Kingdom to provide the epidemic predictions and evaluate the effectiveness of control measures. We divide infectious individuals into asymptomatic and symptomatic infectious individuals. The symptomatic infectious phase is also divided into reported (severe symptoms) and unreported (mild symptoms) cases. In fact, there exists a period for the cumulative number of reported cases to grow (approximately) exponentially in the early phase of virus transmission which is around the implementation of the national prevention and control measures. We firstly combine the date of the implementation of the measures with the daily and cumulative data of the reported confirmed cases to find the most consistent period for the cumulative number of reported cases to grow approximately exponentially with the formula $\chi_1$1 exp(χ_2t) - χ_3, thus we can determine the parameters χ_1, χ_2, χ_3 in this formula and then determine the parameters and initial conditions for our model by using this formula and the plausible biological parameters for SARS-CoV-2 based on current evidence.We then provide the epidemic predictions, evaluate the effectiveness of control measures by simulations of our model.

Interpretation

We used the plausible biological parameters f, ν, η for SARS-CoV-2 based on current evidence which might be refined as more comprehensive data become available. Our prediction also relies on the cumulative data of the reported confirmed cases.

1 Introduction

An epidemic outbreak of a new human coronavirus, termed the novel coronavirus 2019-nCov, has emerged in Wuhan capital city of Hubei province, China at the end of 2019 [1]. Coronaviruses are a group of enveloped viruses with nonsegmented, single-stranded, and positive-sense RNA genomes. Apart from infecting a variety of economically important vertebrates (such as pigs and chickens), six coronaviruses have been known to infect human hosts and cause respiratory diseases. Among them, severe acute respiratory syndrome coronavirus (SARS-CoV) and Middle East respiratory syndrome coronavirus (MERS-CoV) are zoonotic and highly pathogenic coronaviruses that have resulted in regional and global outbreaks. Coronaviruses possess a distinctive morphology, the name being derived from the outer fringe, or "corona" of embedded envelope protein.

The main goal of this paper is to predict forward in time the future number of cases from early reported case data in regions throughout the world. Our models incorporate the following important elements of COVID-19 epidemics: (1) the number of asymptomatic infectious individuals (with no or very mild symptoms), (2) the number of symptomatic reported infectious individuals (with severe symptoms) and (3) the number of symptomatic unreported infectious individuals (with mild symptoms). In this paper, we will show the prediction of the final size of the asymptomatic infectious, reported (with severe symptoms) and unreported cases (with mild symptoms) which is an important epidemiological problem research teams around the world are trying to solve [1] [2] [3] [4] [5] .

In an early phase of the epidemic, the reported case data grows exponentially, which corresponds to a constant transmission rate. We assume that government measures and public awareness cause this early constant transmission rate to change to a time dependent exponentially decreasing rate. Early reports shows that 2019-nCov transmission may occur from an infectious individual, who is not yet symptomatic [2]. Evidently, such asymptomatic infectious cases are not reported to medical authorities. For epidemic influenza outbreaks, reported cases are typically only a fraction of the total number of the symptomatic infectious individuals. For the current epidemic in (various countries), it is likely that intensive efforts by Chinese public health authorities, have reduced the number of unreported cases. We see that estimation of the number of asymptomatic infectious and unreported cases has major importance in understanding the severity of this epidemic. Strong measures are needed to curb mild and asymptomatic cases that are fueling the pandemic.

2 Model

To provide the epidemic predictions, evaluate the effectiveness of control measures, we fit the following transmission dynamic model to the cumulative confirmed case data in China, South Korea, Italy, France, Germany and United Kingdom:

$$\begin{cases} S'(t) = -\tau(t)S(t)[I(t)+U(t)] \\ I'(t) = \tau(t)S(t)[I(t)+U(t)] - \nu I(t) \\ R'(t) = \nu_1 I(t) - \eta R(t) \\ U'(t) = \nu_2 I(t) - \eta U(t) \end{cases} \tag{2.1}$$

With the initial data

$$S(t_0) = S_0 > 0, I(t_0) = I_0, R(t_0) = 0 \quad \text{and} \quad U(t_0) = U_0 \geqslant 0 \tag{2.2}$$

Here $t \geqslant t_0$ is time in days, t_0 is the beginning date in the model of the epidemic, $S(t)$ is the number of individuals susceptible to infection at time t, $I(t)$ is the number of asymptomatic infectious individuals at time t, $R(t)$ is the number of reported sever symptomatic infectious individuals at time t, and $U(t)$ is the number of unreported mild symptomatic infectious individuals at time t [1].

The time-dependent parameter $\tau(t)$ is the transmission rate. During the early phase of the epidemic, when the cumulative number of reported cases grows approximately exponential, is a constant value τ_0. Asymptomatic infectious individuals $I(t)$ are infectious for an average period of $\frac{1}{\nu}$ days. Reported symptomatic individuals $R(t)$ are infectious for an average period of $\frac{1}{\eta}$ days, as are unreported symptomatic individuals $U(t)$. We assume that reported symptomatic infectious individuals $R(t)$ are reported and isolated immediately, and cause no further infections. The asymptomatic individuals $I(t)$ can also be viewed as having a low-level symptomatic state. All infections are acquired from either $I(t)$ or $U(t)$ individuals.

The parameters of the model are listed in 5.1 and schematic diagram of the model is given in

Symbol	Interpretation	Method
t_0	Time at which the epidemic started	fitted
t_0	Number of susceptible at time t_0	fixed
I_0	Number of asymptomatic infectious at time t_0	fitted
U_0	Number of unreported symptomatic infectious at time t_0	fitted
$\tau(t)$	Transmission rate at time t	fitted
$\frac{1}{\nu}$	Average time during which asymptomatic infectious are asymptomatic	fixed
f	Fraction of asymptomatic infectious that become reported symptomatic infectious	fixed
$\nu_1 = f\nu$	Rate at which asymptomatic infectious become reported symptomatic	fitted
$\nu_2 = (1-f)\nu$	Rate at which asymptomatic infectious become unreported symptomatic	fitted
$\frac{1}{\eta}$	Average time symptomatic infectious have symptoms	fixed

Table 2.1: Parameters and initial conditions of the model

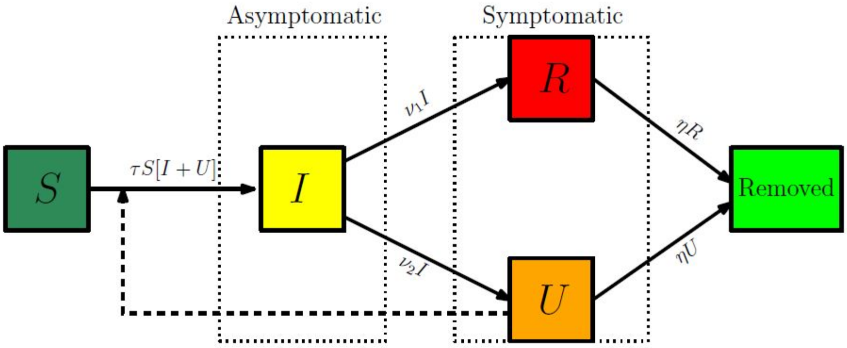
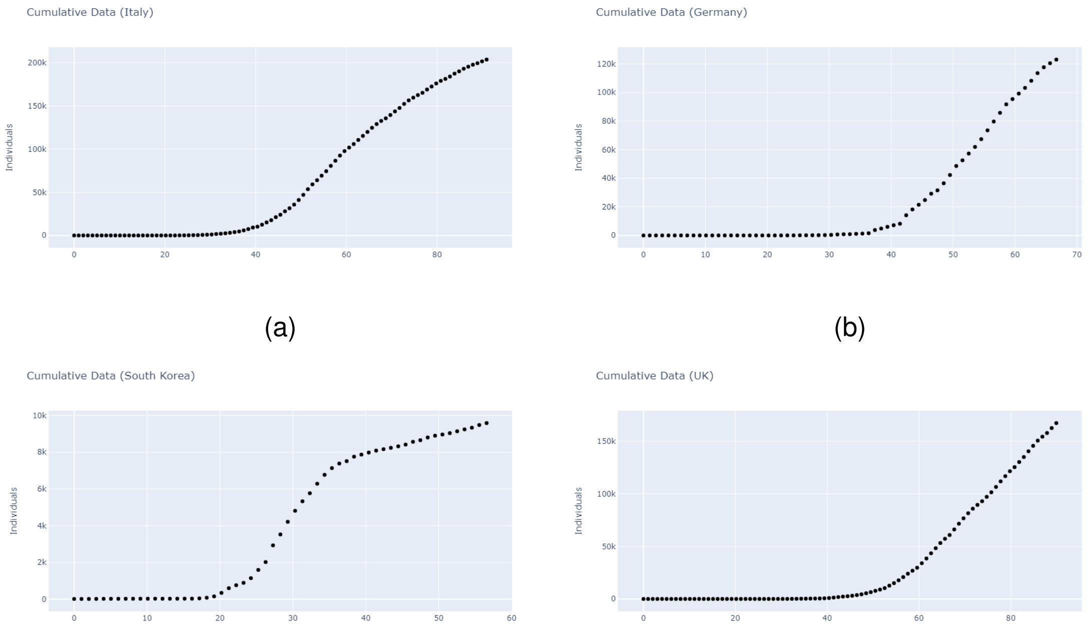


Figure : Compartments and flow chart of the model

2.1 Data

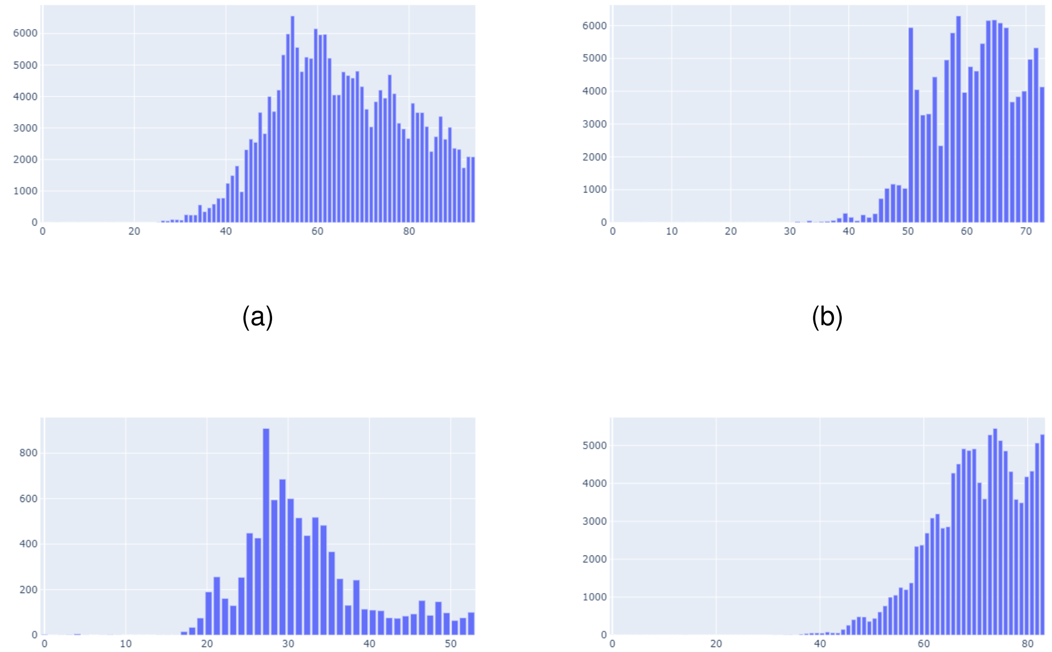
We use the cumulative reported and daily reported data from the Italian ministry of health, the Robert Koch Institute of Germany, Korean center for disease control and Public Health England.

*The below graphs shows the cumulative number reported and daily reported cases for Covid*19 *for* (*a*) *Italy between Febraury* 23 *and March* 03; (*b*) *Germany between Febraury* 24 *and March* 11; (*c*) *South Korea between January* 20 *and March* 09; (*d*) *UK between Febraury* 26 *and March* 29


Cumulative Data (Italy)
Individuals
0
50k
100k
150k
200k
0
20
40
60
80
Cumulative Data (Germany)
Individuals
0
20k
40k
60k
80k
100k
120k
0
10
20
30
40
50
60
70
Cumulative Data (South Korea)
Individuals
0
2k
4k
6k
8k
10k
0
10
20
30
40
50
60
Cumulative Data (UK)
Individuals
0
50k
100k
150k
0
20
40
60
80

(a) (b)

(c) (d)


0
1000
2000
3000
4000
5000
6000
0
20
40
60
80
0
1000
2000
3000
4000
5000
6000
0
10
20
30
40
50
60
70
0
200
400
600
800
0
10
20
30
40
50
0
1000
2000
3000
4000
5000
0
20
40
60
80

(a) (b)

(c) (d)

3 Estimation of the parameters and initial conditions

The parameters t, ν, ν_1, ν_2, η, as well as the starting time t_0 and the initial conditions $S(t_0)$, $I(t_0)$, $U(t_0)$ are uncertain. Our objective is to identify them from specific time data of reported symptomatic infectious cases. To identify the unreported asymptomatic infectious cases, we assume that the cumulative reported symptomatic infectious cases at time t consist of a constant fraction f of the total number of symptomatic infectious cases at time t. In other words, we assume that the removal rate ν of infectious asymptomatic cases $I(t)$ takes the following form: $\nu = \nu_1 + \nu_2$, where $\nu_1 = f\nu$ is the removal rate of reported symptomatic infectious individuals, and $\nu_2 = (1-f)\nu$ is the removal rate of unreported symptomatic infectious individuals due to all causes. The cumulative number of reported symptomatic infectious cases at time t, denoted by $CR(t)$, is

$$CR(t) = \nu_1 \int_t^{t_0} I(\sigma)d\sigma, \quad \text{for} \quad t \geq t_0 \tag{3.1}$$

and the cumulative number of unreported at time t is given by

$$CU(t) = \nu_2 \int_{t_0}^{t} I(\sigma)d\sigma \quad \text{for} \quad t \geq t_0 \tag{3.2}$$

We assume that $f = 0.9$ which means that 10% of symptomatic infectious cases go unreported. The actual value of f is unknown and varies from country to country. We assume $\eta = 1/7$, which means that the average period of infectiousness of both unreported symptomatic infectious individuals and reported symptomatic infectious individuals is 7 days. We assume $\nu = 1/7$ which means that the average period of infectiousness of asymptomatic infectious individuals is 7 days. These values can be modified as further epidemiological information becomes known.

After a period of linear growth, the epidemic of COVID19 starts to enter into a second phase where the cumulative number of reported cases $CR(t)$ grows approximately exponentially. We assume that during this phase cumulative number of reported cases is described by the following phenomenological model:

$$CR(t) = \chi_1 exp(\chi_2 t) - \chi_3, \quad t \geq t_0 \tag{3.3}$$

We evaluate χ_1,χ_2,χ_3 using the reported case data from the above mentioned in section Data. we obtain the model starting time of the epidemic t_0 from 3.3.

$$CR(t_0) = 0 \iff \chi_1 exp(\chi_2 t_0) - \chi_3 = 0 \implies t_0 = \frac{1}{\chi_2}(ln(\chi_3) - ln(\chi_1))$$

During the exponential growth phase of the epidemic, $\tau \equiv \tau_1$ is constant. When strong government measures such as isolation, quarantine, and public closings are implemented, we use an exponential decrease for a time-dependent decreasing transmission rate $\tau(t)$ to incorporate these effects since the actual effects of these measures are complex. The formula for $\tau(t)$ during this phase is

$$\begin{cases} \tau(t) = \tau_1, 0 \leq t \leq N, \\ \tau(t) = \tau_1 exp(-\mu(t-N)), N < t. \end{cases} \tag{3.4}$$

The date N and the value μ are chosen so that the cumulative reported cases in the numerical simulation of the epidemic aligns with the cumulative reported case data after day N when the public measures take effect. In this way we are able to project forward the time-path of the epidemic after the government imposed public restrictions take effect.

The daily number of reported cases from the model can be obtained by computing the solution of the following equation:

$$DR'(t) = \nu_1 I(t) - DR(t), \quad \text{for} \quad t \geq t_0 \quad \text{and} \quad DR(t_0) = DR_0 \tag{3.5}$$

We assume the initial value S_0, corresponds to the population of the region of the reported case data. The value of the susceptible population $S(t)$ is assumed to be only slightly changed by removal of the number of people infected in the beginning of the second phase. The other initial conditions are

In the following we fix f, ν, η.
Step 1: Since ν, f are fixed, we know that

$$\nu_1 = f\nu \quad \text{and} \quad \nu_2 = (1-f)\nu.$$

Step 2: By using equation () and () we obtain

$$CR'(t) = \nu_1 I(t) \iff \chi_1 \chi_2 exp(\chi_2 t) = \nu I(t) \tag{3.6}$$

and

$$\frac{exp(\chi_2 t)}{exp(\chi_2 t_0)} = \frac{I(t)}{I(t_0)},$$

and therefore

$$I(t) = I_0 exp(\chi(t - t_0). \tag{3.7}$$

Moreover by using (3.6) at $t = t_0$

$$I_0 = \frac{\chi_1 \chi_2 exp(\chi_2 t_0)}{f\nu} = \frac{\chi_3 \chi_2}{f\nu}. \tag{3.8}$$

Step 3: In order to evaluate the parameters of the model we assume the initial value S_0 by replacing $S(t)$ corresponding to the population of the region of the reported case data (which is equivalent to neglecting the variation of suspectibles due to the epidemic, which is consistent with the face that $t \to CR(t)$ grows exponentially). Therefore, it remains to estimate τ and η in the following system:

$$\begin{cases} I'(t) = \tau(t) S_0 [I(t) + U(t)] - \nu I(t) \\ U'(t) = \nu_2 I(t) - \eta U(t) \end{cases} \tag{3.9}$$

By using the first equation we obtain

$$U(t) = \frac{1}{\tau S_0} [I'(t) + \nu I(t)] - I(t) \tag{3.10}$$

and therefore by using (3.7) we must have

$$I(t) = I_0 exp(\chi_2 (t - t_0)) \quad \text{and} \quad U(t) = U_0 exp(\chi_2 (t - t_0),$$

By substituting these expression into (3.9) we get

$$\begin{cases} \chi_2 I_0 = \tau(t) S_0 [I_0 + U_0] - \nu I_0 \\ \chi_2 U_0 = \nu_2 I_0 - \eta U_0. \end{cases} \tag{3.11}$$

By dividing the first equation of (3.11) by I_0 we get

$$\chi_2 = \tau S_0 \left[1 + \frac{U_0}{I_0} \right] - \nu$$

and hence

$$\frac{U_0}{I_0} = \frac{\chi_2 + \nu}{\tau S_0} - 1 \tag{3.12}$$

By using the second equation of the (3.11) we obtain

$$\frac{U_0}{I_0} = \frac{\nu_2}{\eta + \chi_2} \tag{3.13}$$

By using (3.12) and (3.13) we obtain

$$\tau = \frac{\chi_2 + \nu}{S_0} \frac{\eta + \chi_2}{\nu_2 + \eta + \chi_2} \tag{3.14}$$

By using (3.13) we can compute

$$U_0 = \frac{\nu}{\eta + \chi_2} I_0 = \frac{(1-f)\nu}{\eta + \chi_2} I_0 \tag{3.15}$$

3.1 Computation of the basic reproductive number R_0

The basic reproduction number, denoted R_0, is the expected number of secondary cases produced, in a completely susceptible population, by a typical infective individual. If R_0<1, then on average an infected individual produces less than one new infected individual over the course of its infectious period, and the infection cannot grow. Conversely, if R_0>1, then each infected individual produces, on average, more than one new infection, and the disease can invade the population. For the case of a single infected compartment, $R_0$is simply the product of the infection rate and the mean duration of the infection. In this section we apply results from [2] [3]. The linearized equation of the infectious part of the system is given by

$$\begin{cases} I'(t) = \tau S_0[I(t) + U(t)] - \nu I(t) \\ U'(t) = \nu_2 I(t) - \eta U(t) \end{cases} \tag{3.16}$$

The corresponding matrix is

$$A = \begin{bmatrix} \tau S_0 - \nu & \tau S_0 \\ \nu_2 & -\eta \end{bmatrix}$$

and the matrix A can be rewritten as

$$A = V - S$$

where

$$V = \begin{bmatrix} \tau S_0 - \nu & \tau S_0 \\ \nu_2 & -0 \end{bmatrix} \text{ and } \quad S = \begin{bmatrix} \nu & 0 \\ 0 & \eta \end{bmatrix}$$

Therefore, the next generation matrix is

$$VS^{-1} = \begin{bmatrix} \frac{\tau S_0}{\nu} & \frac{\tau S_0}{\eta} \\ \frac{\nu_2}{\nu} & 0 \end{bmatrix}$$

which is a Leslie matrix, and the basic reproductive number becomes

$$R_0 = \frac{\tau S_0}{\nu}\left(1 + \frac{\nu_2}{\eta}\right) \tag{3.17}$$

By using (6.8) we obtain

$$R_0 = \frac{\chi_2 + \nu}{S_0} \frac{\eta + \chi_2}{(1-f)\nu + \eta + \chi_2}\left(1 + \frac{(1-f)\nu}{\eta}\right)$$

and by using ν_2 = $(1-f)$ ν we obtain

$$R_0 = \frac{\chi_2 + \nu}{S_0} \frac{\eta + \chi_2}{(1-f)\nu + \eta + \chi_2}\left(1 + \frac{(1-f)\nu}{\eta}\right). \tag{3.18}$$

4 Model Simulation using Python Programming

```
from scipy.integrate import odeint
import numpy as np
import matplotlib.pyplot as plt
import math
import plotly.graph_objects as go
from ipywidgets import interact
import ipywidgets as wg
%matplotlib inline
%%javascript
IPython.OutputArea.prototype._should_scroll = function(lines) {return false;}

# initialize parameters values
def init_param(country,f):
    if country == "Germany":
        x1 = 50.71333;
        x2= 0.197878;
        x3 = 93.3459;
        Mu = 0.05
        #data from Feb 1. until 13 April
        RealDataDR = [2, 1, 1, 2, 0, 0, 1, 1, 0, 0, 0, 2, 2, 0, 0, 0, 0, 0, 0,
         ↪  0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 4, 26, 10, 54, 18, 28, 39, 66,
         ↪  138, 284, 163, 55, 237, 157, 271, 733, 1043, 1174, 1144, 1042,
         ↪  5940, 4049, 3276, 3311, 4438, 2342, 4954, 5780, 6294, 3965, 4751,
         ↪  4615, 5453, 6156, 6174, 6082, 5936,    3677, 3834, 4003, 4974,
         ↪  5323, 4133]
        RealDataCum = [7, 8, 9, 11, 11, 11, 12, 13, 13, 13, 13, 15, 15, 15, 15,
         ↪  15, 15, 15, 15, 15, 15, 17, 21, 47, 57, 111, 129, 157, 196, 262,
         ↪  400, 684, 847, 902, 1139, 1296, 1567, 3795, 4838, 6012, 7156, 8198,
         ↪  14138, 18187, 21463, 24774, 29212, 31554, 36508, 42288, 48582,
         ↪  52547, 57298, 61913, 67366, 73522, 79696, 85778, 91714, 95391,
         ↪  99225, 103228, 108202, 113525, 117658, 120479, 123016]
        S0 = 82.79e6; #number of susceptible at time t0
        N=  43
        v = 1/7;
        Sai = 1/7;
    if country == "Italy":
        x1 =156.321
        x2= 0.20491
        x3 = 306.389
        Mu = 0.175
        #data start from Jan. 28 until April 30.
        RealDataDR = [0, 0, 0, 3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
        ↪  0, 0, 0, 0, 0, 0, 14, 62, 53, 97, 93, 78, 250, 238, 240, 561, 347,
        ↪  466, 587, 769, 778, 1247, 1492, 1797, 977, 2313, 2651, 2547, 3497,
        ↪  2823, 4000, 3526, 4207, 5322, 5986, 6557, 5560, 4789, 5249, 5210,
        ↪  6153, 5959, 5974, 5217, 4050, 4053, 4782, 4668, 4585, 4805, 4316,
        ↪  3599, 3039, 3836, 4204, 3951, 4694, 4092, 3153, 2972, 2667, 3786,
        ↪  3493, 3491, 3047, 2256, 2729, 3370, 2646, 3021, 2357, 2324, 1739,
        ↪  2091, 2086]
        RealDataCum = [3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,
        ↪  3, 3, 3, 17, 79, 132, 229, 322, 400, 650, 888, 1128, 1689, 2036,
        ↪  2502, 3089, 3858, 4636, 5883, 7375, 9172, 10149, 12462, 15113,
        ↪  17660, 21157, 23980, 27980, 31506, 35713, 41035, 47021, 53578,
        ↪  59138, 63927, 69176, 74386, 80539, 86498, 92472, 97689, 101739,
        ↪  105792, 110574, 115242, 119827, 124632, 128948, 132547, 135586,
        ↪  139422, 143626, 147577, 152271, 156363, 159516, 162488, 165155,
        ↪  168941, 172434, 175925, 178972, 181228, 183957, 187327, 189973,
        ↪  192994, 195351, 197675, 199414, 201505, 203591]
        S0 = 60.48e6; #number of susceptible at time t0
        N = 46
        v = 1/27;
        Sai = 1/14;
    if country == "South Korea":
        x1 = 0.624
        x2= 0.294
        x3 = 1;
        Mu = 2.4
        # data start from Feb.2, 2020 until March 29.
        RealDataDR = [3, 0, 1, 2, 5, 1, 0, 1, 2, 1, 0, 0, 0, 0, 1, 1, 1, 15,
        ↪  34, 75, 190, 256, 161, 130, 254, 449, 427, 909, 595, 686, 600, 516,
        ↪  438, 518, 483, 367, 248, 131, 242, 114, 110, 107, 76, 74, 84, 93,
        ↪  152, 87, 147,98,64,76,100]
        RealDataCum = [15, 15, 16, 18, 23, 24, 24, 25, 27, 28, 28, 28, 28, 28,
        ↪  29, 30, 31, 46, 80, 155, 345, 601, 762, 892, 1146, 1595, 2022,
        ↪  2931, 3526, 4212, 4812, 5328, 5766, 6284, 6767, 7134, 7382, 7513,
        ↪  7755, 7869, 7979, 8086, 8162, 8236, 8320, 8413, 8565, 8652, 8799,
        ↪  8897, 8961, 9037, 9137, 9241, 9332, 9478, 9583]
        S0 = 51.47e6; #number of susceptible at time t0(South Korea population)
        N = 27
        v = 1/7;
        Sai = 1/7;
    if country == "United Kingdom":
        x1 = 113.26598
        x2= 0.163222
        x3 = 353.3245671
        Mu = 0.06
        # data from Feb. 13 until April 26
        RealDataDR = [0, 0, 0, 0, 0, 2, 0, 0, 0, 1, 0, 1, 0, 0, 4, 1, 0, 1, 0,
         ↪  0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 2, 5, 4, 8, 12, 5, 22, 40, 55, 56,
         ↪  51, 81, 60, 57, 148, 259, 406, 484, 478, 361, 442, 611, 769, 999,
         ↪  1055, 1255, 1198, 1378, 2338, 2375, 2692, 3087, 3197, 2822, 2858,
         ↪  4273, 4514, 4913, 4868, 4911, 4020, 3592, 5282, 5450, 5131, 4858,
         ↪  4313, 3579, 3489, 4178, 4326, 5065, 5292]
        RealDataCum = [2, 2, 2, 2, 3, 3, 4, 4, 4, 8, 9, 9, 10, 10, 10, 10, 10,
         ↪  10, 10, 10, 10, 10, 11, 11, 11, 13, 18, 22, 30, 42, 47, 69, 109,
         ↪  164, 220, 271, 352, 412, 469, 617, 876, 1282, 1766, 2244, 2605,
         ↪  3047, 3658, 4427, 5426, 6481, 7736, 8934, 10312, 12650, 15025,
         ↪  17717, 20804, 24001, 26823, 29681, 33954, 38468, 43381, 48249,
         ↪  53160, 57180, 60772, 66054, 71504, 76635, 81493, 85806, 89385,
         ↪  92874, 97052, 101378, 106443, 111735, 116691, 121412, 125265,
         ↪  130119, 134879, 140366, 145524, 150494, 154242, 157715, 162421,
         ↪  167150]
        S0 = 66.44e6; #number of susceptible at time t0
        N = 52
        v = 1/30;
        Sai = 1/31;
    v1 = f*v;
    v2 = (1-f)*v;
    t0 = (1/x2) - math.log(x1);           #initial time for the beginning of the
     ↪  early exponential growth phase
    return N, t0, v, Sai, v1, v2, f, x1, x2, x3, Mu, RealDataDR, RealDataCum,
     ↪  S0

# calculate Initial population in the I and U compartments
def Cal_Init_I0U0(S0,x2,f,v1,v2,Sai):
    I0 = x2/(f*(v1+v2));                         #number of asymptomatic
     ↪  infectious at time t0
    U0 = ((1-f)*(v1 + v2) / (Sai + x2)) * I0;    #number of unreported
     ↪  symptomatic infectious at time t0
    return I0, U0, S0

# transmission rate at time t0
def Ti_0(x2, v1, v2, S, Eta):
    return ((x2 + v1 + v2) / S)*((Eta + x2) / (v2 + Eta + x2))


# time dependent transmission rate
def Ti(t, N, x2, v1, v2, S, Sai, Mu):
    if t <= N:
        retval = ((x2 + v1 + v2) / S)*((Sai + x2) / (v2 + Sai + x2))
    else:
        retval = (((x2 + v1 + v2) / S)*((Sai + x2) / (v2 + Sai + x2)))* math.exp(-Mu*(t - N))
    return retval


# transmission rate over time
def Ti_over_time( N, x2, v1, v2, S, Sai, Mu):
    t = np.linspace(0, 100, 100)      # Grid of time points (in days)
    return [Ti(i, N, x2, v1, v2, S, Sai, Mu) for i in range(len(t))]

# basic reproduction number at time t0
def R_0(ti0, v1, v2, S, Eta):
    return ((ti0 * S)/(v1 + v2)) * (1 + (v2 / Eta))

# basic reproduction number over time
def R_0_over_time(t, N, x2, v1, v2, S, eta, Mu):
    tis = [Ti(i, N, x2, v1, v2, S, eta, Mu) for i in range(len(t))]
    r0s = [R_0(tis[i], v1, v2, S, eta) for i in range(len(tis))]
    return r0s

# calculating cumulative reported and unreported
def CumRep(t,t0,dIdt,N,x1,x2,x3):
    if t <= N:
        retval = x1*math.exp(x2*t)-x3
    else:
        retval = dIdt
    return retval

# differential equation
def deriv_Cum(y, t, Ti, v, Sai, v1, v2 ,f, t0, N, x2, Mu,x1):
    S, I, U, CR, CU = y
    dSdt = -Ti(t, N, x2, v1, v2, S, Sai, Mu) * S *(I + U)
    dIdt = Ti(t, N, x2, v1, v2, S, Sai, Mu) * S * (I + U) - v * I
    dUdt = v2 * I - Sai * U
    CR = I
    CR += CumRep(t, t0, dIdt,N,x1,x2,1)
    CU = I + U
    CU += CumRep(t, t0, dIdt,N,x1,x2,1)
    return dSdt, dIdt, dUdt, CR, CU


def Calc_Cum(S0, x2, v, Sai, v1, v2, f, t0, N, Mu,x1):
    # Initial population in the different compartments
    I0, U0, S0 = Cal_Init_I0U0(S0,x2,f,v1,v2,Sai)
    CR0 = 1.0   #number of cumulated reported symptomatic infectious at time t0
    CU0 = 1.0   #number of cumulated unreported symptomatic infectious at time t0

    t = np.linspace(0, 100, 100)      # Grid of time points (in days)
    y0 = [S0, I0, U0, CR0, CU0]     #initial population in different compartments
    print ('I0 = ' + "{:.2f}".format(I0))
    print ('U0 = ' + "{:.2f}".format(U0))
    print('N = ' + str(N))
    # Integrate the SIR equations over the time grid, t.
    ret = odeint(deriv_Cum, y0, t, args=(Ti, v, Sai, v1, v2, f, t0, N, x2, Mu,x1))
    S, I, U, CR, CU= ret.T
    CR = v1 * CR
    CU = v2 * CU
    return t, CR, CU

# differential equation
def deriv_Daily(y, t, Ti, v, Sai, v1, v2 ,f, N, x2, Mu):
    S, I,  U ,DR = y
    dSdt = -Ti(t, N, x2, v1, v2, S, Sai, Mu) * S *(I + U)
    dIdt = Ti(t, N, x2, v1, v2, S, Sai, Mu) * S * (I + U) - v * I
    dUdt = v2 * I - Sai * U
    dDRdt = (v * f * I) - DR
    return dSdt, dIdt, dUdt ,dDRdt

def Calc_Daily(S0, x2, v, Sai, v1, v2, f, RealDataDR, N, Mu):
    # Initial population in the I and U compartments
    I0, U0, S0 = Cal_Init_I0U0(S0,x2,f,v1,v2,Sai)
    DR0 = 1.0   #daily number of reported symptomatic infectious at time t0

    t = np.linspace(0, 100, 100)      # Grid of time points (in days)
    y0 = [S0, I0, U0, DR0]            #initial population in different compartments

    # Integrate the SIR equations over the time grid, t.
    ret = odeint(deriv_Daily, y0, t, args=(Ti, v, Sai, v1, v2,f, N, x2, Mu))
    S, I,  U, DR = ret.T

    return t, DR, RealDataDR

# differential equation
def deriv(y, t, Ti, v, Sai, v1, v2, N, x2, Mu):
    S, I, R, U  = y
    dSdt = -Ti(t, N, x2, v1, v2, S, Sai, Mu) * S *(I + U)
    dIdt = Ti(t, N, x2, v1, v2, S, Sai, Mu) * S * (I + U) - v * I
    dRdt = v1 * I - Sai * R
    dUdt = v2 * I - Sai * U
    return dSdt, dIdt, dRdt, dUdt
```

```

def Calc_All(S0, x2, v, Sai, v1, v2, f, N, Mu):
    # Initial population in the I and U compartments
    I0, U0, S0 = Cal_Init_I0U0(S0,x2,f,v1,v2,Sai)
    R0 = 1.0   # number of reported symptomatic infectious at time t0

    t = np.linspace(0, 100, 100)      # Grid of time points (in days)
    y0 = [S0, I0, R0, U0]             #initial population in different compartments

    # Integrate the SIR equations over the time grid, t.
    ret = odeint(deriv, y0, t, args=(Ti, v, Sai, v1, v2, N, x2, Mu))
    S, I, R, U = ret.T
    return t, S, I, R, U

def plotsiru(Country, f):
    #initialize parameters based on the country
    N, t0, v, eta, v1, v2, f, x1, x2, x3, Mu, RealDataDR, RealDataCum, S0 = init_param(Country,f)

    #Cumulative graph
    figCum = go.Figure()
    #solving differential equations
    t, CR, CU = Calc_Cum(S0, x2, v, eta, v1, v2, f, t0, N, Mu, x1)
    figCum.add_trace(go.Scatter(x=t, y=CR, name='CR(t)', mode = "lines", line = dict(color = "blue", shape = "spline", smoothing = 1)))
    figCum.add_trace(go.Scatter(x=t, y=CU, name='CU(t)', mode = "lines", line = dict(color = "green", shape = "spline", smoothing = 1)))
    figCum.add_trace(go.Scatter(x=t, y=RealDataCum, name='Cumulative Data', mode = "markers", line = dict(color = "black", shape = "spline", smoothing = 1)))
    figCum.update_layout(autosize=False, width= 900, height=500, title='Cumulative Data ({})'.format(Country), yaxis_title='Individuals')

    #Daily reported cases graph
    figDaily = go.Figure()
    t, DR, RealDataDR = Calc_Daily(S0, x2, v, eta, v1, v2, f, RealDataDR, N, Mu)
    figDaily.add_trace(go.Scatter(x=t, y=DR, name='DR(t)', mode = "lines", line = dict(color = "purple", shape = "spline", smoothing = 1)))
    figDaily.add_trace(go.Scatter(x=t, y=RealDataDR, name='Daily Data', mode = "markers", line = dict(color = "black", shape = "spline", smoothing = 1)))
    #figDaily.add_trace(go.Bar(x=t, y=RealDataDR, name='Daily Data'))
    figDaily.update_layout(autosize=False, width= 900, height=500,title='Daily Data ({})'.format(Country), yaxis_title='Individuals')


    # epidemic curve graph
    figAll = go.Figure()
    t, S, I, R, U = Calc_All(S0, x2, v, eta, v1, v2, f, N, Mu)
    figAll.add_trace(go.Scatter(x=t, y=I, name='I(t)', mode = "lines", line = dict(color = "Red", shape = "spline", smoothing = 1)))
    figAll.add_trace(go.Scatter(x=t, y=R, name='R(t)', mode = "lines", line = dict(color = "blue", shape = "spline", smoothing = 1)))
    figAll.add_trace(go.Scatter(x=t, y=U, name='U(t)', mode = "lines", line = dict(color = "green", shape = "spline", smoothing = 1)))
    figAll.update_layout(autosize=False, width= 900, height=500, title='Epidemic Curve ({})'.format(Country), yaxis_title='Individuals')

    #transmission rate over time
    ti_overtime = Ti_over_time(N, x2, v1, v2, S0, eta, Mu)
    figTi = go.Figure()
    figTi.add_trace(go.Scatter(x=t, y=ti_overtime, name='CR(t)', mode = "lines", line = dict(color = "blue", shape = "spline", smoothing = 1)))
    figTi.update_layout(autosize=False, width= 900, height=500, title='Transmission Rate ({})'.format(Country))

    ti0 = Ti_0(x2, v1, v2, S0, eta)
    print('Ti_0 = ' + "{:.2E}".format(ti0))
    print('R_0 = ' + "{:.2f}".format(R_0(ti0,v1, v2, S0, eta)))
    figR0.show()
    figTi.show()
    figCum.show()
    figDaily.show()
    figAll.show()

    # basic reproduction number over time

    R_0_overtime = R_0_over_time(R0_ti_t, N, x2, v1, v2, S0, eta, Mu)
    figR0 = go.Figure()
    figR0.add_trace(go.Scatter(x=R0_ti_t, y=R_0_overtime, name='Basic reproduction number (t)', mode = "lines", line = dict(color = "green", shape = "spline", smoothing = 1)))
    figR0.update_layout(autosize=False, width= 900, height=500, title='Basic reproduction number over time ({})'.format(Country))

RepFraction = wg.FloatSlider(description='f:',value=0.1,min=0.1, max=0.9,step=0.1)
interact(plotsiru ,Country = ['Germany','Italy','South Korea','United Kingdom'], f= RepFraction)
```

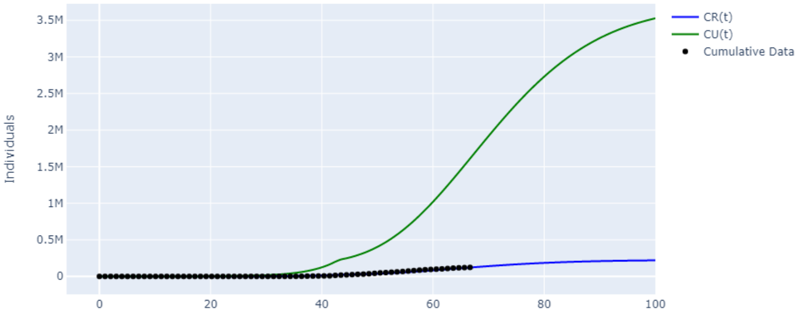
5 Numerical Simulation

In this section we apply method described in (3) to calculate the parameters. The numerical simulations are presented in the chronological order of appearance for four countries Germany, Italy, South Korea and United Kingdom.

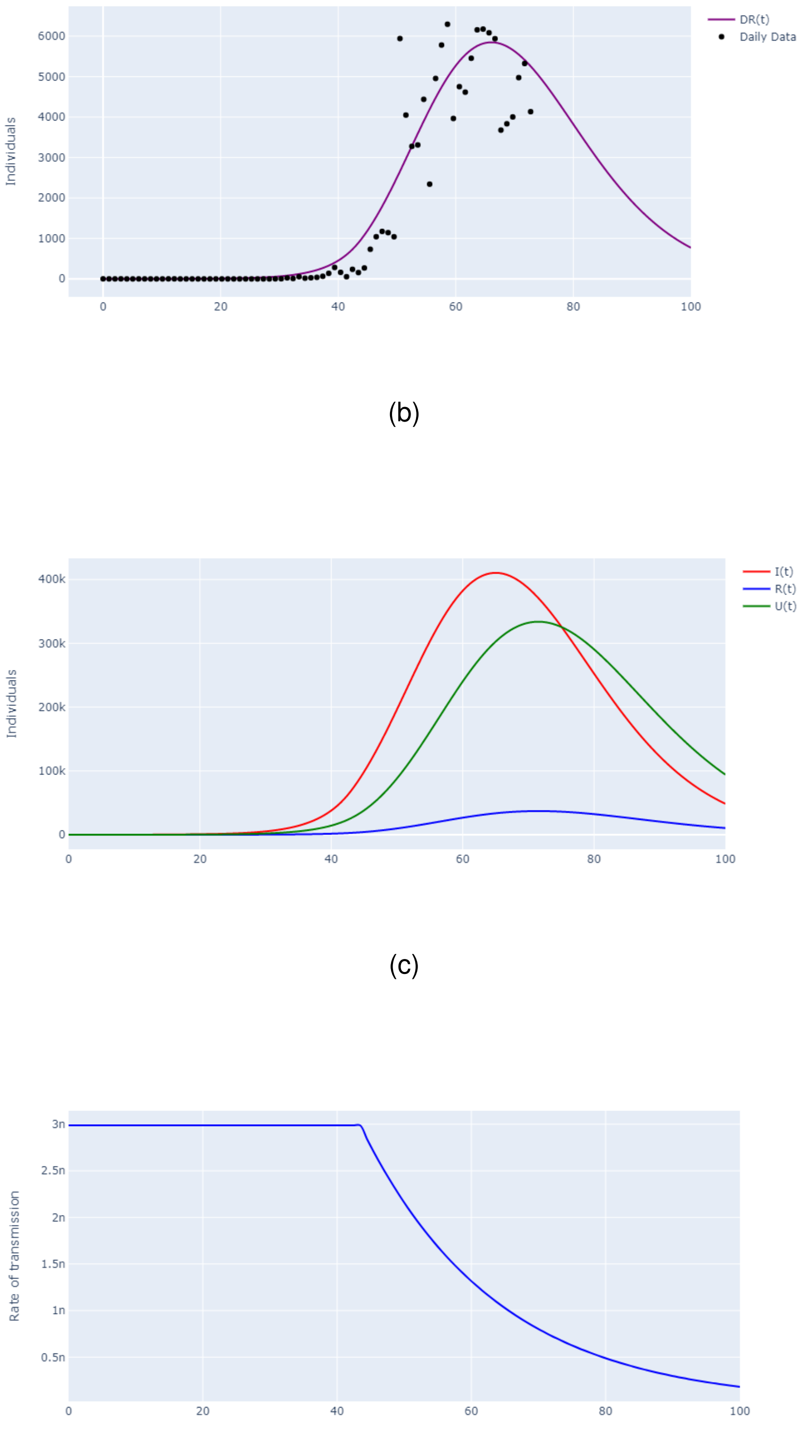
$Country$	χ_1	χ_2	χ_3	t_0	μ	N	S_0	f	τ_0
Germany	50.71333	0.197878	93.3459	Feb 24	0.05	March 11	82.79×10^6	0.1	2.988129n
Italy	156.321	0.20491	306.389	Feb 23	0.175	March 10	60.48×10^6	0.1	3.569836n
South Korea	156.321	0.20491	306.389	Feb 23	0.175	March 10	60.48×10^6	0.1	3.569836n
United Kingdom	113.26598	0.163222	353.3245671	Feb 26	2.4	March 29	66.44×10^6	0.1	2.564777n

Table 5.1: The parameters are calculated according to Data mentioned in section (2.1)

5.1 Predicting the number of cases for Germany



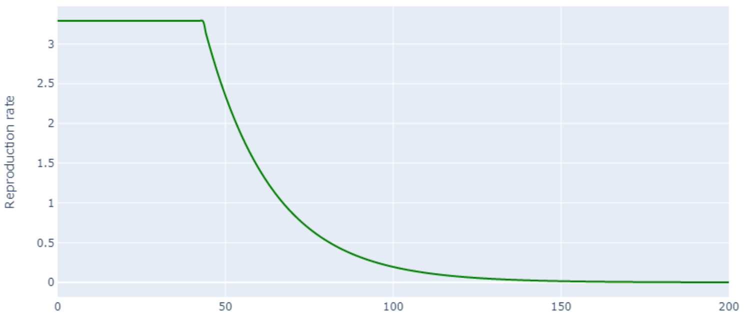
(a)


DR(t)
Daily Data
Individuals
6000
5000
4000
3000
2000
1000
0
0
20
40
60
80
100
I(t)
R(t)
U(t)
Individuals
400k
300k
200k
100k
0
0
20
40
60
80
100
Rate of transmission
3n
2.5n
2n
1.5n
1n
0.5n
0
20
40
60
80
100

(b)

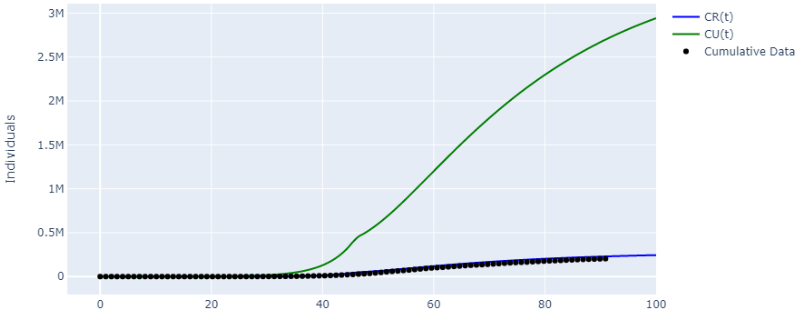
(c)

(d)

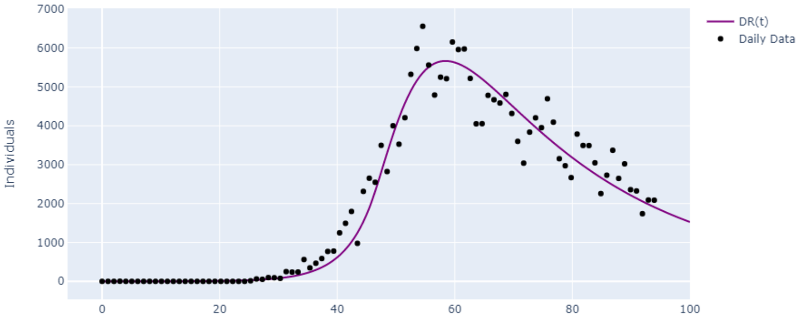


(e)

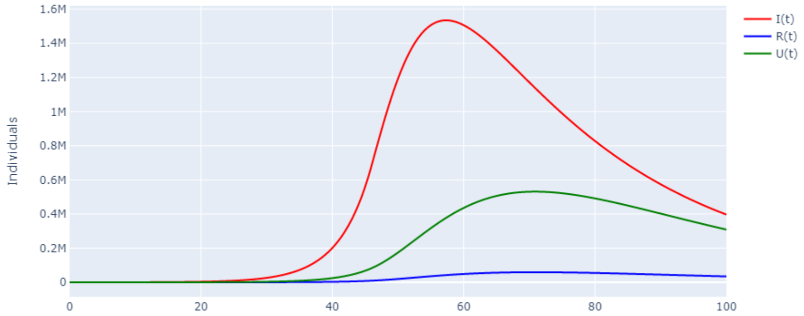
5.2 Predicting the number of cases for Italy



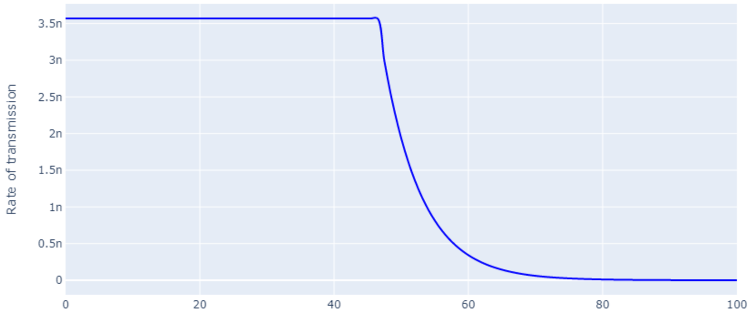
(a)



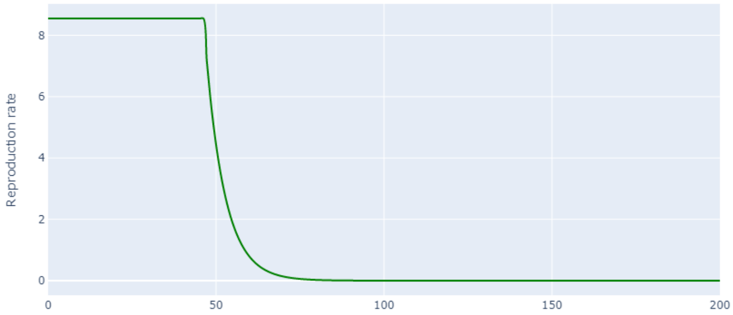
(b)


1.6M
1.4M
1.2M
1M
0.8M
0.6M
0.4M
0.2M
0
Individuals
0
20
40
60
80
100
I(t)
R(t)
U(t)

(c)

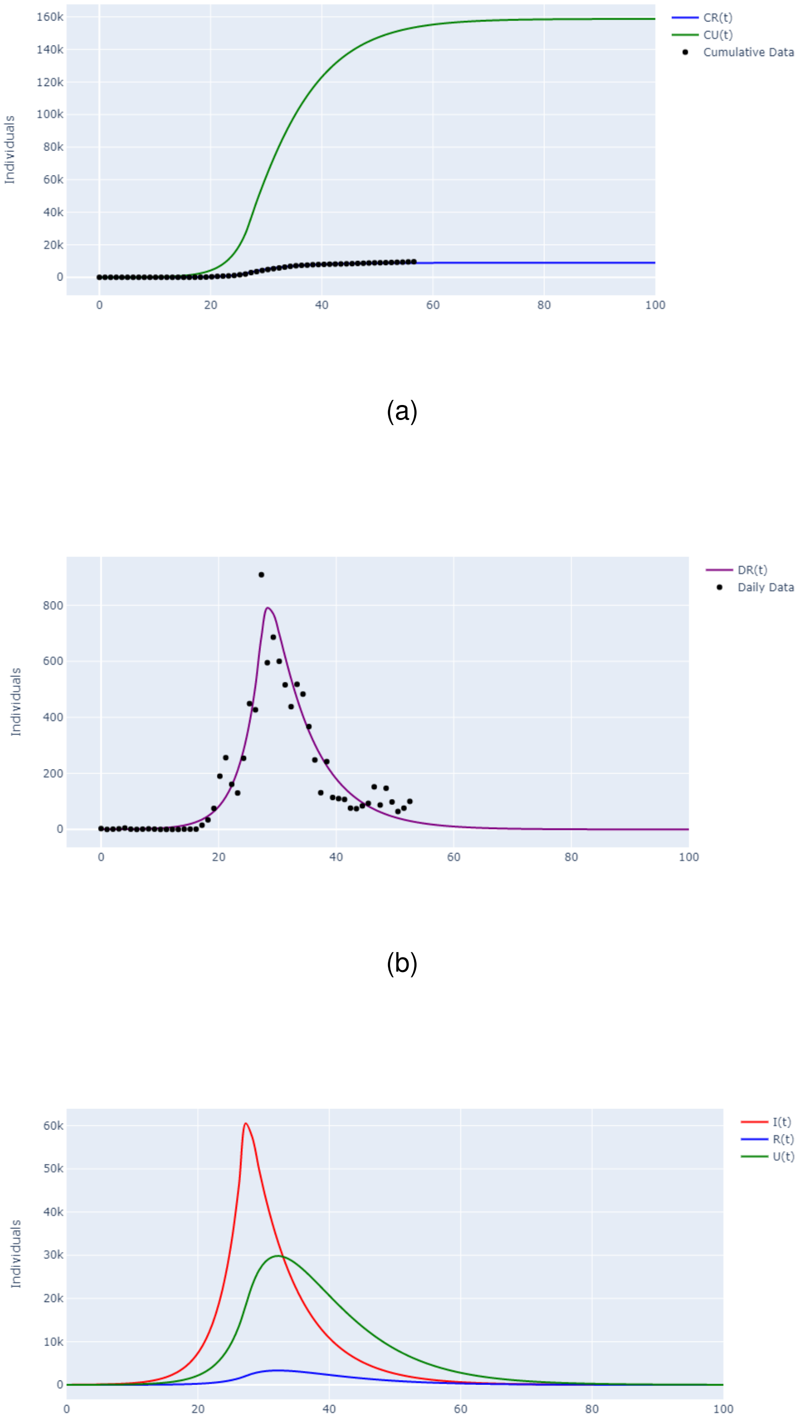

3.5n
3n
2.5n
2n
1.5n
1n
0.5n
0
Rate of transmission
0
20
40
60
80
100

(d)


8
6
4
2
0
Reproduction rate
0
50
100
150
200

(e)

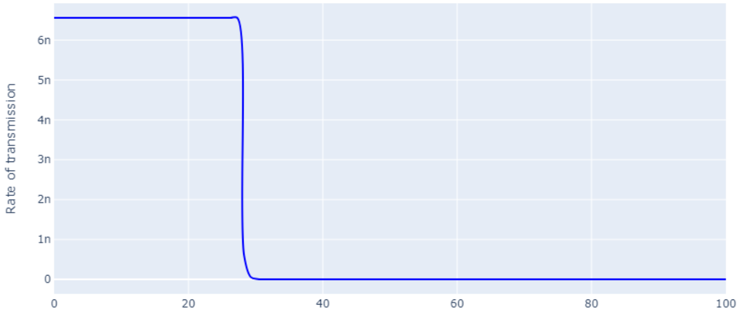
5.3 Predicting the number of cases for South Korea



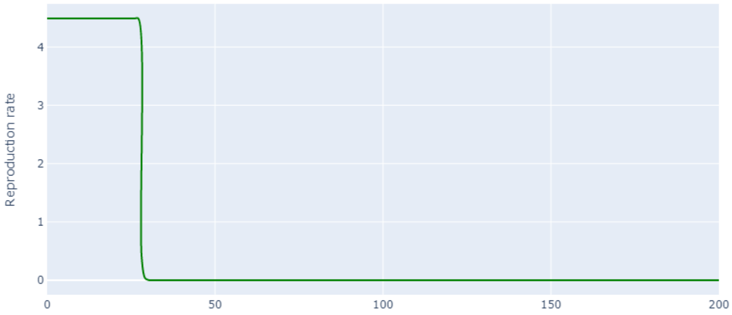
(a)

(b)

(c)

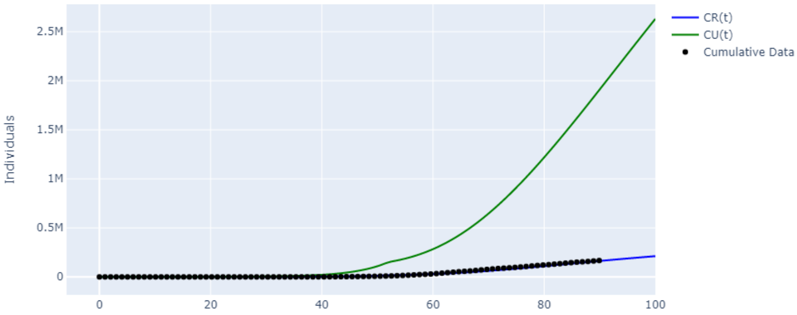


(d)

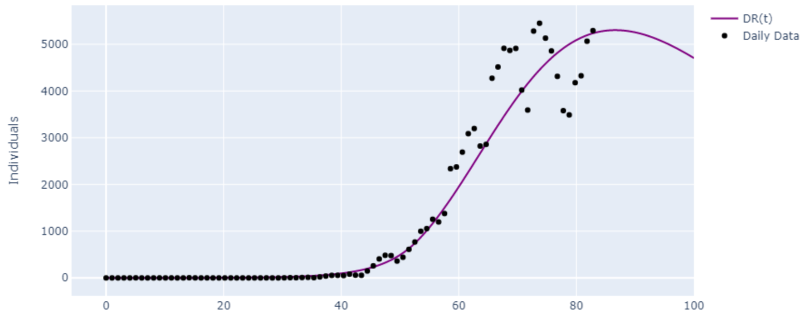


(e)

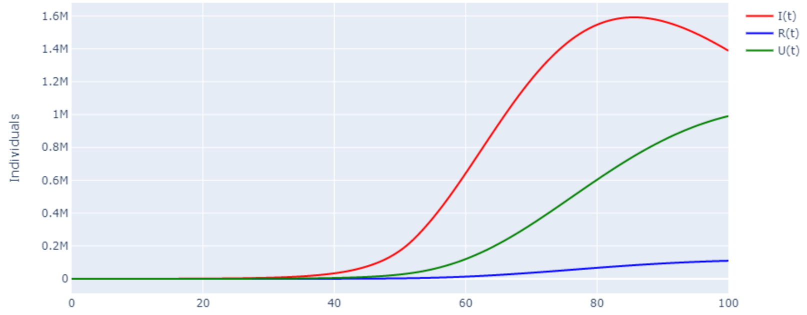
5.4 Predicting the number of cases for United Kingdom



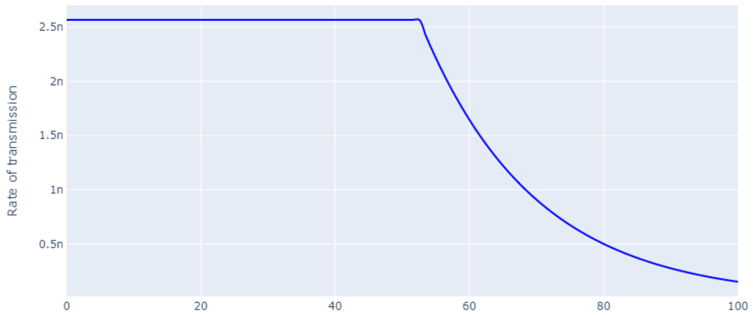
(a)


DR(t)
Daily Data
Individuals
5000
4000
3000
2000
1000
0
0
20
40
60
80
100

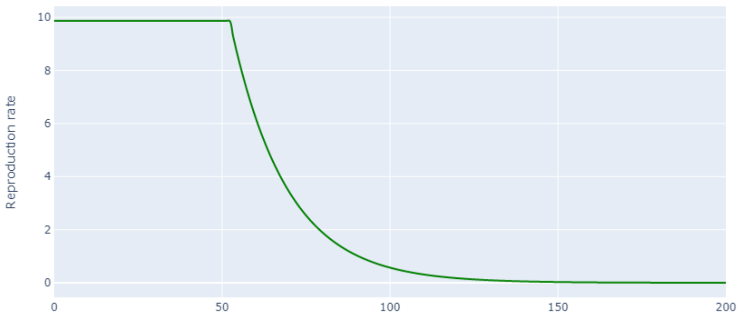
(b)


I(t)
R(t)
U(t)
Individuals
1.6M
1.4M
1.2M
1M
0.8M
0.6M
0.4M
0.2M
0
0
20
40
60
80
100

(c)


Rate of transmission
2.5n
2n
1.5n
1n
0.5n
0
20
40
60
80
100

(d)



(e)

6 Authors Contribution

fnkdn

Bibliography

[1] Pierre magal and Glenn Webb. Predicting the number of reported and unreported cases for the COVID-19 epidemic in south korea, italy, france and germany. mar 2020.

[2] O. Diekmann, J.A.P. Heesterbeek, and J.A.J. Metz. On the definition and the computation of the basic reproduction ratio r 0 in models for infectious diseases in heterogeneous populations. *Journal of Mathematical Biology*, 28(4), jun 1990.

[3] P. van den Driessche and James Watmough. Reproduction numbers and sub-threshold endemic equilibria for compartmental models of disease transmission. *Mathematical Biosciences*, 180(1-2):29–48, nov 2002.

[4] Zhihua Liu, Pierre Magal, Ousmane Seydi, and Glenn Webb. Understanding unreported cases in the COVID-19 epidemic outbreak in wuhan, china, and the importance of major public health interventions. *Biology*, 9(3):50, mar 2020.

[5] Zhihua Liu, pierre magal, Ousmane Seydi, and Glenn Webb. Predicting the cumulative number of cases for the COVID-19 epidemic in china from early data. mar 2020.