DOI: 10.37943/12QWEH5166

Indira Uvaliyeva
PhD, Associate Professor of the Department of Mathematical and Computer Modeling
iuvalieva@mail.ru, orcid.org/0000-0002-2117-5390
D. Serikbayev East Kazakhstan Technical University, Kazakhstan

Saule Belginova
PhD, Associate Professor of the Department of Information Technology
sbelginova@gmail.com, orcid.org/0000-0002-7238-6016
Turan University, Kazakhstan

Sanzhar Sovetbekov
Master's student of the Department of Mathematical and Computer Modeling
sanzhar.sovetbekov@mail.ru
D. Serikbayev East Kazakhstan Technical University, Kazakhstan

MATHEMATICAL AND COMPUTER MODELS OF THE COVID-19 EPIDEMIC

Abstract: The COVID-19 epidemic has gone down in history as an emergency of international importance. Currently, the number of people infected with coronavirus around the world continues to grow, and modeling such a complex system as the spread of infection is one of the most pressing problems. Various models are used to understand the progress of the COVID-19 coronavirus epidemic and to plan effective control strategies. Such models require the use of advanced computing, such as artificial intelligence, machine learning, cloud computing, and edge computing. This article uses the SIR mathematical model, which is often used and simple to model the prevalence of COVID-19 infection. The SIR model can provide a theoretical basis for studying the prevalence of the COVID-19 virus in a specific population and an understanding of the temporal evolution of the virus. One of the main advantages of this model is the ease of adjusting the sampling parameters as the study scale increases and the most appropriate graphs between the data and the resulting assumptions. Computer models based on the mathematical SIR model of the spread of the COVID-19 epidemic make it possible to estimate the number of possible deaths in the future. In addition, on the basis of the proposed models, it will be possible to assess the effectiveness of measures taken to prevent infection by comparing published data with forecasts. Computer models in Python are created on the basis of the proposed mathematical apparatus of SIR. The following libraries were added in the Python high-level programming language for the numerical solution of the system of differential equations for the SIR model: NumPy, Matplotlib PyPlot and the Integrate module from the SciPy library.

Keywords: coronavirus, COVID-19, mathematical model, SIR model, computer model, Python, epidemic model.

Scientific Journal of Astana IT University
ISSN (P): 2707-9031 ISSN (E): 2707-904X
VOLUME 12, DECEMBER 2022

Introduction

The COVID-19 epidemic has gone down in history as an international emergency. At present, the number of people infected with coronavirus continues to increase day by day around the world and modeling a complex system such as the spread of infection is one of the most pressing problems. The prevention, diagnosis, and treatment of a new coronavirus infection [1], as well as the etiology, epidemiology, and clinic of Covid-19 have been studied [2]. In [3], the authors describe in detail the features of the clinical course, the possibilities of diagnosis, treatment, and prevention of infection in adults and children. Modeling the spread of infection is one of the effective ways to predict the epidemic situation in society. In order to determine the best strategies to reduce the impact of the COVID-19 epidemic, all countries of the world and the World Health Organization are widely using simulations of the spread of this infection. Basically, these are epidemiological models aimed at studying the spread of the disease and the impact of various interventions [4].

The purpose of the research developed within the framework of the scientific article is to create a computer model based on a mathematical model that is well adapted to the COVID-19 epidemic, considering the main characteristics, allowing predicting the spread of the epidemic.

Research objectives:

- Study of indicators of the spread of the COVID-19 epidemic;
- definition of a mathematical model that calculates the spread of the epidemic;
- creation of a computer model of the spread of the COVID-19 epidemic based on a mathematical model.

The novelty of the study lies in the application of the SIR mathematical model for modeling the spread of the COVID-19 disease and its software implementation in the Python language. The software implementation makes it possible to visualize the mathematical model for detecting the primary case of infection, considering epidemiological features.

Materials and research methods. Coronavirus is a whole group of viruses, covering more than 30 species [5]. The species are grouped into 2 subfamilies. They can be transmitted not only to people but also to animals – cats, dogs, birds, pigs and cattle. The virus was discovered in 1960. It got its name because of the villi on the bark, which move in different directions and resemble a crown (Figure 1). Coronaviruses can cause a range of illnesses, from the simple flu to severe acute respiratory syndrome (SARS or «atypical pneumonia»).

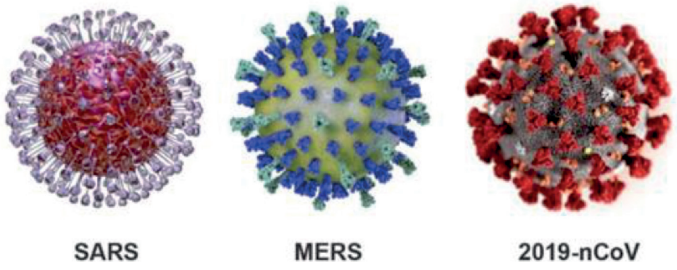

Figure 1. Types of coronavirus

To summarize what is known about the coronavirus and the current outbreak, the study conducted a literature review of publicly available materials. In work [6], the protein of coronavirus, its life cycle, non-structural proteins, symptoms, and techniques for doing in-silico research on COVID, its relationship with the neurological system, detection strategies, and current treatments are all discussed. In the paper [7], the authors provide reports and analyses based on data that describe an important and methodologically sound vulnerability

DOI: 10.37943/12QWEH5166

map for this disease. Authors of [8] summarize current knowledge about COVID-19, such as the transmission process, diagnostic methods, clinical signs, pathological characteristics, and treatment measures. In the [9], a group of scientists provides a detailed overview of the disease Covid-19.

COVID-19 is short for the words «coronavirus disease» («disease caused by coronavirus»). In this name, «CO» means «corona» (corona), «VI» – «virus» (virus), «D» – « disease «(disease), and «19» is the year when the disease was first discovered. And the virus that causes the disease COVID-19 is called SARS-CoV-2. The name is given because the virus is genetically identical to the SARS-CoV virus [2].

The dynamic nature of the COVID-19 epidemic is critical to the implementation of effective response or prevention measures. Most people in general become vulnerable to infection in the early stages of an epidemic [3]. The spread of disease from person to person can be modeled as a stochastic «branching process». If an infected person infects an average of two people, the number of people infected at each stage doubles, and this process grows exponentially. Figure 2 graphically describes the process of spreading the virus.

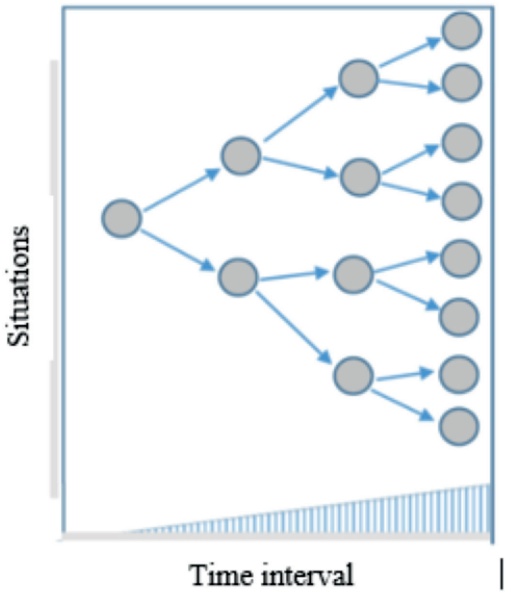

Figure 2. The process of infection spread

According to official data, as of 1.01.2022, 280 989 264 people with the COVID-19 coronavirus were registered in 226 countries over the entire period. The percentage of people who were infected with COVID-19, treated for infection, and died is shown in Figure 3.

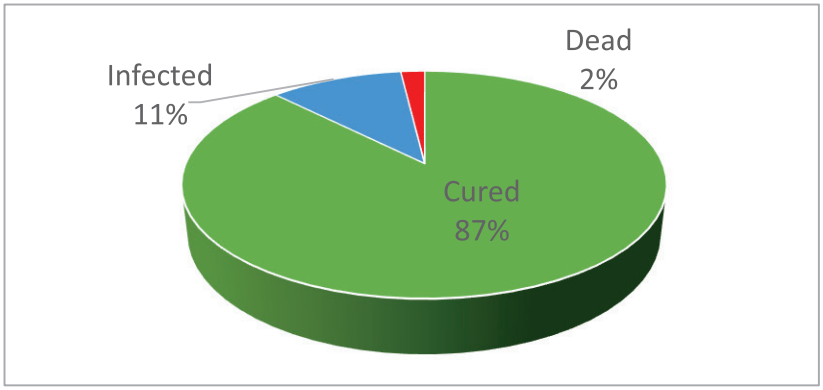

Figure 3. Indicators of the infection spread

Scientific Journal of Astana IT University
ISSN (P): 2707-9031 ISSN (E): 2707-904X
VOLUME 12, DECEMBER 2022

There are 29 654 212 people in the active phase of the coronavirus infection. The total number of deaths from coronavirus is 5 339 847 people, which is 2% of all cases. The number of people who have completely recovered from the COVID-19 coronavirus worldwide is 245 995 205. This is 87% of all infected people.

Mathematical models of the spread of the COVID-19 virus epidemic

The mathematical modeling of diseases is an urgent problem in the modern world. Many researchers resort to mathematical models to predict a particular disease [4], since they help to investigate changes correctly and accurately in certain processes occurring in society. Mathematical modeling is necessary in certain areas of medicine, for example, in epidemiology, where specific experiments are impossible or difficult [10].

The principles of mathematical models in the epidemiology of diseases differ from the models used in other sciences, since there are no certain invariable dependencies proven by practice [11]. In order to correctly model an epidemic, in addition to the appropriate basic model, it is necessary to choose its correct quantitative parameters. Some of these parameters are related to the biology of the virus. Other parameters are specific to a particular city or region and should be selected based on local data.

In the early stages of an epidemic, its spread occurs exponentially, i.e., with every N Day (the worse the situation, the smaller N), the number of patients' doubles [12]. This means that the doubling period gradually increases over time, which means that the exponential function best describes an epidemic only in its early stages and in the short term.

COVID-19 is an acute global problem due to its contagious nature and frequently changing characteristics. The rapid growth of this disease has forced scientists to take urgent countermeasures to stop the epidemic. Scientists have proposed and implemented various technologies to reduce the negative impact of the pandemic and speed up the recovery period. The authors of the article [13] describe a model whose results for the most important epidemiological parameters, such as the number of active cases, cumulative deaths, daily new deaths and daily new cases (among others), are consistent with the available real data on the first and subsequent waves of COVID-19. Since mathematical modeling of the COVID-19 epidemic based on system dynamics and SIR models is considered inadequate. The authors of work 14 propose a non-classical discipline of epidemic dynamics to overcome the shortcomings of modeling [14]. The article [15] provides a brief overview of the results of the classical SIR model and variants that consider contact frequency heterogeneity. It notes that calibrating a classical model against data generated by a heterogeneous model can lead to forecasts that are biased for several reasons and underestimate forecast uncertainty [15]. These technologies include artificial intelligence; machine learning; deep learning; fog computing; internet tools; cognitive computing; wireless connection.

Modeling of the coronavirus spread prediction based on data from the beginning of 2020 was made by the authors of the works. For example, the authors of the work provide a simulation of the spread of the disease on real data in the conditions of March 2020 [16]. The article [17] examines the SEIAR (Susceptible, Exposed, Infected, Symptomatic, Asymptomatically Infected and Recovered) COVID-19 infection model with a constant advection rate for disease spread. In [18] proposed a mathematical model to describe the dynamics of the spread of the Middle East respiratory syndrome.

Smart health services require formal processes to assess, prevent, and manage the spread of COVID-19. These technologies can be used to reduce the negative impact of this pandemic and speed up the treatment period for infection. The following models were used to study the spread of the virus in the population:

- susceptible-infected (SI);

DOI: 10.37943/12QWEH5166

- susceptible-infected-cured (SIR);
- sensitive-incubation period-infected-cured (SEIR);
- susceptible-infected-susceptible (SIS);
- susceptible-infected-cured-susceptible (SIRS) models.

These models offer two possible outcomes. First, if new infections are not monitored, the disease could eventually become an epidemic. The second result is that if the necessary measures are taken to protect susceptible people from infection, the virus will die. COVID-19 disease is the same as other infectious diseases and requires contact tracing to reduce new infections.

The basic model for characterizing the spread of an epidemic, the SIR model, which is considered the «gold standard» of epidemiology, was first proposed by the Scottish epidemiologists Kermac and Mackendrick in the 1920s. The SIR model aims to predict the number of people who are susceptible to infection, actively infected, or cured of infection at any given moment [19]. The article [20] provides mathematical and numerical analysis to solve important questions on the COVID-19 disease. Reviewed studies allow the models to predict disease behavior by evaluating a small number of parameters. There, each person in the study population can be in one of three cases:

- Susceptible (susceptible to infection);
- Infected;
- Removed (cured of infection and not susceptible to infection).

It is assumed that people can be immune to infection only after complete treatment for infection, which means that the sequence Susceptible → Infected → Removed can pass for each person at any time.

The SIR model is designed to solve many problems related to the evolution of the spread of the virus in real time, both quantitatively and qualitatively in a useful way. This is a dynamic system represented by three related codes describing the temporal evolution of the following three populations [21]:

- Sensitive people: These are people who are not infected, but may become infected. As the virus spreads, a large number of people infect it, so that the sensitive population increases over a certain period of time (growth period).
- Infected people: These are people who are infected with the virus and can transmit it to sensitive people. An infected person can remain in this population, and he will be removed from the infected population in case of his treatment or death.
- Cured people: These are people who have recovered (or died) from the virus.

The general scheme of a person's transition from one state to another is shown in Figure 4.

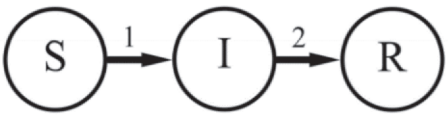

Figure 4. The general scheme of a person's transition from one state to another

The order of the main assumptions when using this SIR model is as follows [22]:

- each of the susceptible people who have been in contact with an infected person has a certain probability of infection, regardless of time;
- the level of infection is proportional to the number of infected people, as well as the number of people susceptible to infection;
- each infected person, in turn, has a constant probability that he will be cured in a certain unit of time;
- the rate of cure for an infection is proportional to the number of infected people.

Scientific Journal of Astana IT University
ISSN (P): 2707-9031 ISSN (E): 2707-904X
VOLUME 12, DECEMBER 2022

As shown in the graph shown in Fig. 5, if the number of infected people during a certain period of time exceeds the number of people cured of the infection, this means that this period is characterized by the spread of the epidemic. Otherwise, we can say that the epidemic will disappear.

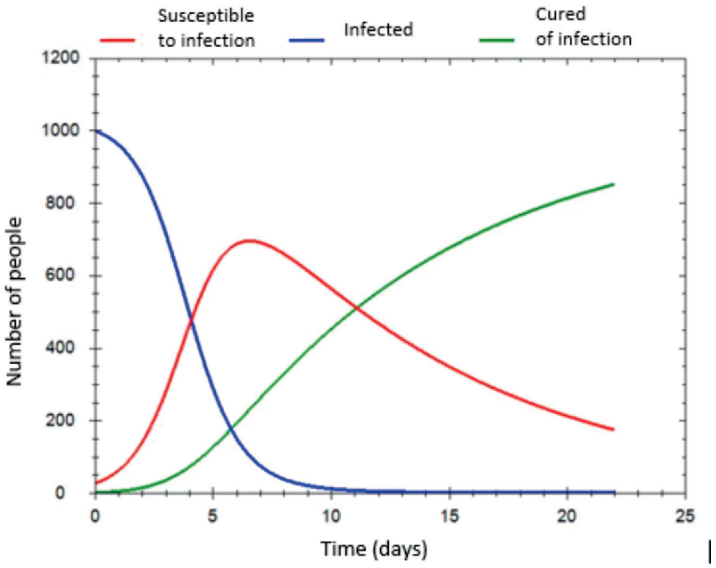

Figure 5. Graph of the SIR model

It is obvious that before the outbreak of the epidemic, 100% of people were in group S (susceptible to infection) and no data was recorded in the remaining groups. For convenience, we calculate the population of 100 people, respectively $S = 100$, $I = 0$, $R = 0$. In such cases, of course, the epidemic does not spread, since there must be at least one patient to start it. Therefore, consider the following case: $S = 99$, $I = 1$, $R = 0$. Now the epidemic is spreading, and its modeling consists in sequentially calculating the state of the population at the next step [20].

To understand how many people are infected at every step, it is important to understand the presence of two probabilities: the probability of contact between two people and the probability of infection of an infected person by contact with a sensitive person (β). Most often, $1/N$ (N is the volume of the population) is used in the model to represent the first probability, which means that each person contacts one random person in the population every time. The second probability (β) provides an accurate biological indicator of the infection of a particular pathogen (with all the influencing factors: temperature, the presence of a mask, etc.).

An infected person infects a person at a certain time, sensitive to a particular infection, when meeting with such a probability (1):

$$\frac{\beta}{N} \tag{1}$$

Then he will infect all sensitive people as follows (2):

$$\frac{S*\beta}{N} \tag{2}$$

All infected together infect susceptible people (3):

DOI: 10.37943/12QWEH5166

$$\frac{S*I*\beta}{N} \tag{3}$$

Thus, at the second step of the model, the number of people susceptible to infection decreases by $0.99 * \beta$.

In addition, with a certain probability (γ), the infected, which are considered as the number of the inverse time of the disease, are cured of the disease. If the disease lasts 10 days, then the patient is treated on a certain day with a probability $\gamma = 1/10$. At each step, the number of cured of the infection will be equal to the following equation (4):

$$I * \gamma \tag{4}$$

At the second stage, the number of infected people decreases by $1 * \gamma$ and also increases by $0.99 * \beta$. The number of people cured of the infection is increased by $1 * \gamma$. The resulting relative state is calculated by the next step of the model. Thus, the model is formulated using the following system of equations (5):

$$\begin{cases} \frac{dS(t)}{dt} = -\frac{\beta*I(t)*S(t)}{N} \\ \frac{dI(t)}{dt} = \frac{\beta*I(t)*S(t)}{N} - \gamma * I(t) \\ \frac{dR(t)}{dt} = \gamma * I(t) \end{cases} \tag{5}$$

where: $S(t)$ – the number of people susceptible to infection per unit of time t; $I(t)$ is the number of infected people per time unit t; $R(t)$ is the number of people treated for infection per time unit t; β is the coefficient of intensity of human infection; γ–intensity coefficient of treatment of infected persons; N is the total population size.

The first equation of system (5) means that the change in the number of healthy people (as well as those susceptible to disease) decreases over time in proportion to the number of contacts with infected people. After contact, infection occurs, a person sensitive to the disease passes to the infected.

The second equation of system (5) shows that the growth rate of the number of infected people increases in proportion to the number of contacts between healthy and infected people and decreases as the latter are treated for the disease.

The third equation of system (5) shows that the number of people cured of a disease per unit of time is proportional to the number of infected people. In other words, each sick person should be treated after a certain time [23].

In general, the SIR model makes it possible to estimate at least the approximate dynamics of the spread of the epidemic in a first approximation. But the real process of the disease is somewhat complicated, and a number of factors must be taken into account when modeling. First, this disease process can consist of at least two stages: an incubation period (without external signs of the disease) and a period of immediate disease (with external signs of the disease and isolation of the infected person) [24]. In addition, the disease of each person can take different forms: Mild (when a person can carry the disease at home), moderate (when there is a need for hospitalization) and severe (in the COVID-19 example, the disease turns into coronavirus pneumonia, when there is a need to transfer patients to the intensive care unit and use ventilators).

Scientific Journal of Astana IT University
ISSN (P): 2707-9031 ISSN (E): 2707-904X
VOLUME 12, DECEMBER 2022

Results and its discussion. Examples of visualization of the spread of Covid-19 in Python based on the SIR mathematical model**.**

The following libraries were added in the Python high-level programming language for the numerical solution of the system of differential equations for the SIR model: NumPy, Matplotlib PyPlot and the Integrate module from the SciPy library [25]. Now the function that determines the right-hand side of the system of differential equations, indicated above, is called sir_model. This brings up solutions that use this function in the following. Therefore, there are certain agreements on the parameters of this function. The function of the IR model, which calculates the right-hand side of the system of equations, is shown in Figure 6.

t and y are the first two required parameters of this function. t is the value of the moment of time required to calculate the right-hand side of the system. the y parameter is a NumPy array with three lengths. Each component of this array stores the value of the functions S, I and R per time unit t. For convenience, we simply write these values as S, I, R = y. In addition, this function uses three additional parameters. These are: N (total population size), beta (speed of spread of β-infection) and gamma (rate of treatment of γ-infection).

```
def sir_model(t, y, N, beta, gamma):
    S, I, R = y
    dS_dt = -beta*I*S/N
    dI_dt = beta*I*S/N - gamma*I
    dR_dt = gamma*I
    return dS_dt, dI_dt, dR_dt
```

Figure 6. Construction of a function that calculates the right side of the system of equations of the SIR model

To calculate the right value of the system according to the first equation of formula (1), the right value dS_dt=-beta*i*S/N is set, according to the second equation dI_dt = beta*i*s/N-gamma*I and the right value of the third equation is denoted dR_dt = gamma*I. This function must return a string, list, or NumPy array of length three. We do this with the return statement. So, a function is written that calculates the right-hand side of the system. Thus, the input values of the function are the time t, the values of the equations S, I and R, as well as the parameters N, beta, and gamma. The function returns the right-hand side of the system according to the written formulas.

The SIR model describes the change in the population of each of these compartments in terms of two parameters, β and γ. β describes the effective contact rate of the disease: an infected individual comes into contact with βN other individuals per unit time (of which the fraction that are susceptible to contracting the disease is S/N). γ is the mean recovery rate: that is, $1/\gamma$ is the mean period of time during which an infected individual can pass it on [26].

Next, we enter the parameters and initial conditions. Parameters: N, beta, and gamma. Enter the value of each parameter that belongs to it. Initial conditions: I0 (number of infected people at initial time), R0 (number of people cured of infection at initial time) and S0=N-I0-R0 (number of people susceptible to remaining infection). Total population size (N) be equal to 200 000 (the population of the city), speed of spread of β-infection (beta) is 0.32 (the period of autumn infections) and rate of treatment of γ-infection (gamma) is be 0.12.

Enter the value of each parameter that belongs to it. Initial conditions: I0 (number of infected people at initial time), R0 (number of people cured of infection at initial time) and S0=N-I0-R0 (number of people susceptible to remaining infection).

DOI: 10.37943/12QWEH5166


```
N = 200000
beta = .32
gamma = .12
I0 = 100
R0 = 0
S0 = N - I0 - R0
```

Figure 7. Entering parameters and initial conditions

t enters the time and the solve_ivp solution is used. IVP (initial value problem) – solution of a differential equation with an initial condition. The Solve_ivp function is located in the integrate module. First of all, we can create a wrapper function (wrapper function). Such functions are called lambda functions. The lambda function depends on two parameters y and T. The Solve_ivp function contains arguments. The first argument of this function is the name of the function that computes the right side of the system. In this case, it is sir_model. The following sir_model parameters are now displayed: t, y, N, beta, gamma. As the second argument, we must specify the integration interval that is looking for a solution. For example, from 0 to T. The third argument is the initial conditions S0, I 0, R0. These are the three required arguments for solving solve_ivp. Finally, the value True is used for the dence_output argument. The result of the code is shown in Figure 8.

```
T = 150
integrate.solve_ivp(lambda t, y: sir_model (t, y, N, beta, gamma), [0, T], [S0, I0, R0], dense_output = True)
```

```
Out[4]:  message: 'The solver successfully reached the end of the integration interval.'
            nfev: 134
            njev: 0
             nlu: 0
             sol: <scipy.integrate._ivp.common.OdeSolution object at 0x0000026F9B4F2910>
          status: 0
         success: True
               t: array([0.00000000e+00, 1.17850541e-04, 1.29635595e-03, 1.30814100e-02,
                1.30931951e-01, 1.30943736e+00, 5.47165982e+00, 1.13956670e+01,
                1.82531365e+01, 2.57452888e+01, 3.45396943e+01, 4.38557221e+01,
                5.30866788e+01, 6.35899303e+01, 7.51060437e+01, 8.66221572e+01,
                9.74715548e+01, 1.08142159e+02, 1.18752353e+02, 1.29340528e+02,
                1.39920507e+02, 1.50000000e+02])
```

Figure 8. Solver call

The resulting corresponding solution is stored in the sol variable, as shown in the following figure. Sol is a solver, i.e. a function created by solve_ivp. It allows you to calculate all the necessary equations S, I and R from the value of T. The code for saving the solution to the variable sol is shown in Figure 9.

```
T = 150
sol = integrate.solve_ivp(lambda t, y: sir_model (t, y, N, beta, gamma), [0, T], [S0, I0, R0], dense_output = True)
```

Figure 9. Saving the solution to the sol variable

The next step is to visualize the solution. To do this, you first need an array of time units t = np.linspace(). Then, using the sol.sol function, calculate the S, I, and R equations for all points in the array t: y = sol.sol(t). To get the answer to each equation, we write them like this: S, I, R = y[0,:], y = [1,:], y = [2,:]. Thus, the S, I, and R equations are represented by a graph function. The result of the solution visualization is shown in Figure 10.

Scientific Journal of Astana IT University
ISSN (P): 2707-9031 ISSN (E): 2707-904X
VOLUME 12, DECEMBER 2022

```
t = np.linspace(0, T, 300)
y = sol.sol(t)
S, I, R = y[0, :], y[1, :], y[2, :]
plt.plot(t, S, label='Susceptible')
plt.plot(t, I, label='Infected')
plt.plot(t, R, label='Recovered')
plt.xlabel('t')
plt.legend()
plt.grid()
pass
```

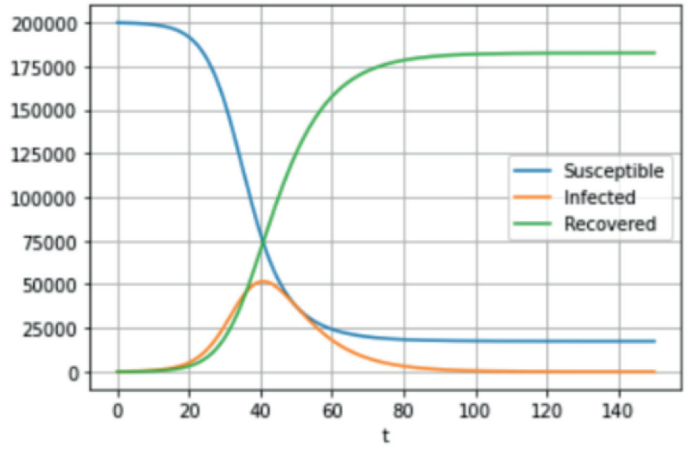

Figure 10. Visualization of the solution

You can change the input parameters of a task using the interaction function from the IpyWidgets module of the Python programming language. The sliders that change the value of the S, I, and R equations have a def react_sir function with beta and gamma arguments. The interaction function has three arguments. The first is the name of the function (interact_sir), the next two arguments are defined as beta and gamma. The interactive activation code is shown in Figure 11.

```
from ipywidgets import interact

def interact_sir(beta = .32, gamma = .12):

    T = 150
    sol = integrate.solve_ivp(lambda t, y: sir_model (t, y, N, beta, gamma), [0, T], [S0, I0, R0], dense_output = True)
    t = np.linspace(0, T, 300)
    y = sol.sol(t)
    S, I, R = y[0, :], y[1, :], y[2, :]
    plt.plot(t, S, label='Susceptible')
    plt.plot(t, I, label='Infected')
    plt.plot(t, R, label='Recovered')
    plt.xlabel('t')
    plt.legend()
    plt.grid()


interact(interact_sir, beta=(0, 2, .02), gamma=(0, .5, .01))
```

Figure 11. Interactive connection code

An example of a software implementation of a mathematical model for the spread of the COVID-19 epidemic in Python is shown in Figure 12.

DOI: 10.37943/12QWEH5166

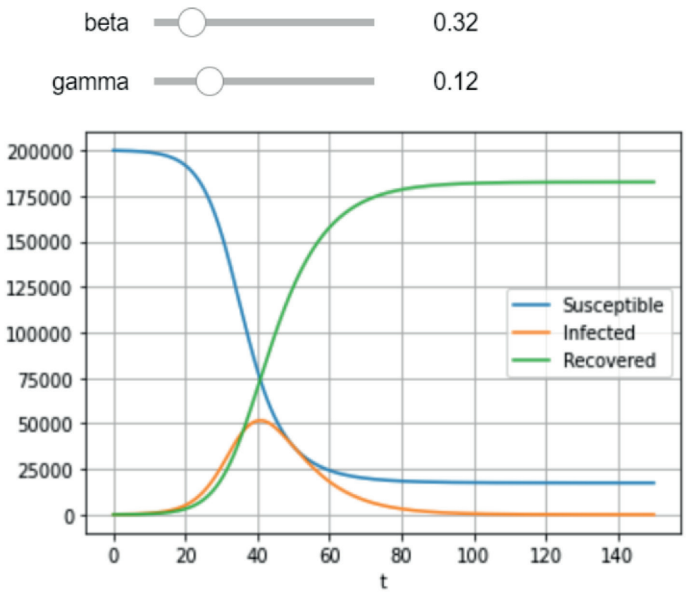

Figure 12. Results of the work

Conclusion

The article describes the results of a study regarding the modeling of the COVID-19 epidemic. The presented model highlights the important role of direct human-to-human transmission of COVID-19 infection. On the other hand, the presented model does not take into account the characteristics of the infection, social distancing, the duration of contact with an infected person, as well as important factors influencing the extent of the infection. The use of this model plays an important role in determining the peak of infection and the risk of infection. A model for detecting a primary case of infection has been developed and visualized, taking into account the epidemiological characteristics of the infection. Analysis of the factors influencing the number of infections allows the development of specific models in future studies.

References

1. Nikiforov, V.V., Suranova, T.G., Mironov, A. YU., & Zabozlayev F.G. (2020). Novaya koronavirusnaya infektsiya (COVID-19): etiologiya, epidemiologiya, klinika, diagnostika, lecheniye i profilaktika [New coronavirus infection (COVID-19): etiology, epidemiology, clinic, diagnosis, treatment and prevention]. *Internist. Moskva*. https://internist.ru/publications/detail/novaya-koronavirusnaya-infektsiya-covid-19-etiologiya-epidemiologiya-klinika-lechenie-i-profilaktika/
2. Starshinova, A.A., Kushnareva, E.A., Malkova, A.M., Dovgalyuk, I.F., & Kudlay D.A. (2020). Novaja koronavirusnaja infekcija: osobennosti klinicheskogo techenija, vozmozhnosti diagnostiki, lechenija i profilaktiki infekcii u vzroslyh i detej. Voprosy sovremennoj pediatrii. [New coronaviral infection: Features of clinical course, capabilities of diagnostics, treatment and prevention in adults and children]. *Voprosy sovremennoj pediatrii. 19*(2):123-131. https://doi.org/10.15690/vsp.v19i2.2105
3. Ivanova, A.A., Potashev, A.V., Potasheva, Ye.V. (2016). Matematicheskoye modelirovaniye dinamiki razvitiya epidemiy [Mathematical modeling of epidemic dynamics]. *Nauchnoye obozreniye. 24*, 269-272.
4. Matveyev, A.V. (2020). Matematicheskoye modelirovaniye otsenki effektivnosti mer protiv rasprostraneniya epidemii COVID-19 [Mathematical modeling of the effectiveness of measures against the spread of the COVID-19 epidemic]. *Natsional'naya bezopasnost' i strategicheskoye planirovaniye. 1*, 23-39.

Scientific Journal of Astana IT University
ISSN (P): 2707-9031 ISSN (E): 2707-904X
VOLUME 12, DECEMBER 2022

5. Navjot, V., & Monica (2022). Coronavirus: A review. *Journal of Emerging Technologies and innovative research (JETIR), 9*(3), 1-11. https://www.researchgate.net/publication/360312247_CORONAVIRUS_A_REVIEW
6. Lewis, K. (2022). Understanding youth disconnection in the age of coronavirus. *Community Quality-of-Life Indicators*. Springer, 105-123. https://doi.org/10.1007/978-3-031-06940-6_7
7. Mandeep, S., Dhruv, D. (2022). A recent review on: Coronavirus disease 2019. *Asian Journal of Pharmaceutical and Clinical Research*. Published by Innovare Academic Sciences. https://doi.org/10.22159/ajpcr.2022.v15i7.44547
8. Chavan, M.B., Tarade, D., Pagare, K.H., & Jain, R.S. (2021). A review on coronavirus. *Research Journal of Science and Technology*, *13*(4), 253-255. https://doi.org/10.52711/2349-2988.2021.00039
9. Briggs, W.M. (2022). The most infamous coronavirus forecast. In *International Conference of the Thailand Econometrics Society* (pp. 39-49). Springer, Cham. https://doi.org/10.1007/978-3-030-97273-8_4
10. Rodkin M.V., Shikhova N.M. (2020). Matematicheskoye modelirovaniye razvitiya epidemii COVID-19, popytka prognoza [Mathematical modeling of the COVID-19 epidemic, an attempt at prediction.]. *Ural'skiy geologicheskiy zhurnal*, *3*, 3-13]
11. Thomas, D.M., Sturdivant, R., Dhurandhar, N.V., Debroy, S., & Clark, N. (2020). A primer on COVID-19 mathematical models. *Obesity*, *28*(8), 1375-1377. https://doi.org/10.1002/oby.22881
12. Chen, X., Li, J., Xiao, C., & Yang, P. (2021). Numerical solution and parameter estimation for uncertain SIR model with application to COVID-19. *Fuzzy Optimization and Decision Making, 20*(2), 189-208. https://doi.org/10.1007/s10700-020-09342-9
13. Muñoz-Fernández, G.A., Seoane, J.M., & Seoane-Sepúlveda, J.B. (2021). A SIR-type model describing the successive waves of COVID-19. *Chaos, Solitons & Fractals*, 144, 110682. https://doi.org/10.1016/j.chaos.2021.110682
14. Comunian, A., Gaburro, R., & Giudici, M. (2020). Inversion of a SIR-based model: a critical analysis about the application to COVID-19 epidemic. *Physica D: Nonlinear Phenomena*, 413, 132674. https://doi.org/10.1016/j.physd.2020.132674
15. Ellison, G. (2020). Implications of heterogeneous SIR models for analyses of COVID-19. *National Bureau of Economic Research*, 27373. http://www.nber.org/papers/w27373.pdf
16. Townsend, C. (2022). Modeling Coronavirus-19. In *A Risky Business* (pp. 345-371). Palgrave Macmillan, Cham. https://doi.org/10.1007/978-3-031-11673-5_17
17. Azam, S., Nauman, A., Raza, A., Muhammad Sajid, I., Rafiq, M., Khan, I.,... & Iqbal, Z. (2021). Numerical analysis of novel coronavirus (2019-nCov) pandemic model with advection. *Computers, Materials, & Continua*, 2933–2953. https://doi.org/10.32604/cmc.2021.012396
18. DarAssi, M. H., Shatnawi, T. A., & Safi, M. A. (2022). Mathematical analysis of a MERS-Cov coronavirus model. *Demonstratio Mathematica*, *55*(1), 265-276. https://doi.org/10.1515/dema-2022-0022
19. Cooper, I., Mondal, A., & Antonopoulos, C. G. (2020). A SIR model assumption for the spread of COVID-19 in different communities. *Chaos, Solitons & Fractals*. 139, 110057. https://www.sciencegate.app/document/10.1016/j.chaos.2020.110057
20. Chen, Y.C. et al. (2020). A time-dependent SIR model for COVID-19 with undetectable infected persons. *IEEE Transactions on Network Science and Engineering*, 7(4), 3279-3294. http://gibbs1.ee.nthu.edu.tw/A_TIME_DEPENDENT_SIR_MODEL_FOR_COVID_19.PDF
21. Gavrilina A.V., Sokolov S.V. (2018). Analiz SIR-modeli rasprostraneniya zabolevaniy [Analysis of the SIR model of disease transmission]. *Protsessy upravleniya i ustoychivost'*, *5*(1), 229-232.
22. Moein, S., Nickaeen, N., Roointan, A., Borhani, N., Heidary, Z., Javanmard, S. H.,... & Gheisari, Y. (2021). Inefficiency of SIR models in forecasting COVID-19 epidemic: a case study of Isfahan. *Scientific reports*, *11*(1), 1–9. https://doi.org/10.1038/s41598-021-84055-6
23. Comunian A., Gaburro R., & Giudici M. (2020). Inversion of a SIR-based model: a critical analysis about the application to COVID-19 epidemic. *Physica D: Nonlinear Phenomena*, *413*, 132674. https://doi.org/10.1016/j.physd.2020.132674
24. Lee, G. et al. (2019). PyWavelets: A Python package for wavelet analysis. *Journal of Open Source Software, 4*(36), 1237. https://doi.org/10.21105/joss.01237
25. Vallat, R. (2018). Pingouin: statistics in Python. *Journal of Open Source Software, 3*(31), 1026. https://doi.org/10.21105/joss.01026