Identification and prediction of time-varying parameters of COVID-19 model: a data-driven deep learning approach

Jie Long[a], A. Q. M. Khaliq[b] & K. M. Furati[c]

[a,b]Department of Mathematical Sciences, Middle Tennessee State University, Murfreesboro, TN 37132, USA; [b]Department of Mathematics and Statistics, King Fahd University of Petroleum and Minerals, Dhahran 31261, Saudi Arabia

ABSTRACT
Data-driven deep learning provides efficient algorithms for parameter identification of epidemiology models. Unlike the constant parameters, the complexity of identifying time-varying parameters is largely increased. In this paper, a variant of physics-informed neural network (PINN) is adopted to identify the time-varying parameters of the Susceptible-Infectious-Recovered-Deceased model for the spread of COVID-19 by fitting daily reported cases. The learned parameters are verified by utilizing an ordinary differential equation solver to compute the corresponding solutions of this compartmental model. The effective reproduction number based on these parameters is calculated. Long Short-Term Memory (LSTM) neural network is employed to predict the future weekly time-varying parameters. The numerical simulations demonstrate that PINN combined with LSTM yields accurate and effective results.

1. Introduction

The novel SARS-CoV-2 (severe acute respiratory syndrome coronavirus 2) has spread all over the world since its discovery at the end of 2019, with millions of confirmed cases and more than a million deaths [29]. On March 11, 2020, COVID-19 was characterized as a pandemic by the World Health Organization (WHO) [28]. Due to the huge negative effects of COVID-19, precautionary measures have been aggressively carried out worldwide, such as facial masking, contact tracing, social distancing, and some governmental actions such as lockdowns. Hence, it is significant to analyze the dynamics of COVID-19 so that the effectiveness of those implemented measures can be verified.

Epidemiological models provide an efficient tool for determining and explaining the dynamics of disease transmission. In the early stage, one of the classical models is the Susceptible-Infectious-Recovery (SIR) model presented by Kermack and McKendrick in 1927 [13]. This compartmental model computes the theoretical number of suscep-

Jie Long. Email: jl7u@mtmail.mtsu.edu

A. Q. M. Khaliq. Email: abdul.khaliq@mtsu.edu

K. M. Furati. Email: kmfurati@kfupm.edu.sa

tible people that became infected by the disease and then recovered. Based on the SIR model, many models have been proposed for different diseases, like SARS and COVID-19 [24, 8, 27]. The Susceptible-Exposed-Infectious-Removed (SEIR) model is proposed to analyze the effect of the precautionary measures on the dynamics of the epidemic [2]. Taking quarantine into account, a SEIQR model is investigated analytically and numerically [19]. The SEIR model has been extended by two extra classes of populations namely, 'C', which is the number of cumulative cases, and 'D' that is the number of severe, critical, and deceased cases, respectively, to understand the trends of COVID-19 outbreak in Wuhan, China [15].

The SIRD model is governed by three parameters β, γ, and μ. Commonly, these parameters are assumed as independent of time during the epidemic. However, the interventions and measures implemented by governments to control the spread of COVID-19 result in significantly varying parameters with time. Thus, if parameters are still considered as constants, the epidemiology model cannot discover the varied dynamic of this disease transmission that is affected by those efficient precautions. Regarding parameters as time-series enable the model to react to the changes in real situations.

Recently, data-driven algorithms were developed to the determination of such time-varying parameters. Raissi et al. [21] proposed PINN to approximate parameters, which are constants, of the SIR model using a small data set. Furthermore, Wang et al. [26] divided the period time into four phases so that their parameters become piecewise constants. Also, there are some studies relating to the time-dependent parameter, like a time-dependent SIR model is proposed to track their parameter time-series [3]. Similarly, the time-dependent parameters of the SIR model were identified by utilizing a neural network to analyze the COVID-19 spread in South Korea [11].

Predicting the trend of COVID-19 transmission is a significant and arduous topic. A model of COVID-19 in China was developed to predict the cumulative number of reported cases [16]. By averaging the slope of approximated parameters of the SIRD model over the last several days, the extrapolated trends of infected, recovered, deceased, and susceptible cases were obtained via solving the model with estimated parameters [18]. An artificial intelligence method was used to predict the pandemic by training on the 2003 SARS data [30]. Recently, several forecasting models including autoregressive integrated moving average, support vector regression, LSTM, and bidirectional LSTM (BiLSTM) were proposed to predict the confirmed cases, deaths, and recoveries [23]. Similarly, Recurrent Neural Network (RNN), LSTM, BiLSTM, Gated recurrent units (GRUs), and Variational AutoEncoder (VAE) algorithms were employed to forecast the number of infected and recovered cases [31].

Based on data collected, the SIRD model is used in this study to simulate the dynamics of the disease. Parameters of this model are regarded as daily and weekly time-varying. As for daily time-varying parameters, it represents that parameters would be varied from each day so that they are more similar to real situations. However, the difficulty of parameter identification increases with the smaller time interval by utilizing some general regression methods such as least square methods. PINN is still able to approximate these time-series parameters with some changes in architecture. Then, they are verified by being substituted into the SIRD model, and then a numerical algorithm is used to solve this ordinary differential equation (ODE) system.

When it comes to weekly time-series, we divide the collected data week by week and employ the interpolation method for each week to gain more data so that the neural network has enough information to learn. The neural network implemented here is the PINN, which can estimate parameters to fit the collected data [20]. When the iden-

tification of these weekly time-series parameters is validated, LSTM is implemented to predict future parameters. In this way, by examining weekly values instead of daily ones, LSTM can still accurately predict future parameters. Another reason to predict the parameters rather than forecast the infectious cases directly is that we could obtain corresponding reproduction number R_0. It is an essential threshold to reflect the disease transmission speed and verify the effectiveness of these measures in curbing the outbreak of COVID-19.
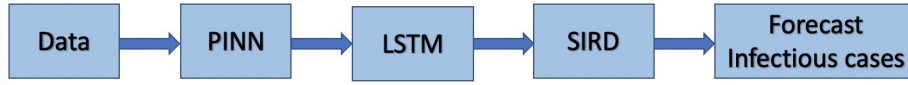
Figure 1. Flow Chart of Data-Driven Algorithm

In Figure 1, we describe that PINN approximates the parameters of the SIRD model by training collected data. Then, LSTM is proposed to predict the future value of parameters. After that, obtained parameters are substituted into the SIRD model so that we could predict the infectious cases.

Outline of the paper: In Section 2 we introduce the SIRD model and explain model parameters (β, γ, and μ). Following that, we introduce the effective reproduction number and how to compute it. Deep learning neural networks, including feedforward neural network (FNN), PINN, and LSTM, are described briefly in Section 3. In Section 4, the collected data is described, and the loss function of PINN is formulated. After that, we present LSTM neural networks for the future prediction of parameters. The simulation results are discussed in Section 5.

2. Epidemiology Model

In this section, we introduce the SIRD model employed in our study. One promise of the SIRD model is that natural birth and death rates are neglected or equivalent so that the total population is considered as constant. Then, the population could be divided into four mutually exclusive groups, which are susceptible, infected, recovered, and deceased, respectively. Also, the reproduction number is introduced in this section.

2.1. SIRD Model

We consider the SIRD model described by the following system of ordinary differential equations [5]:

$$
\begin{aligned}
\dot{S}(t) &= -\frac{\beta(t)S(t)I(t)}{N} \\
\dot{I}(t) &= \frac{\beta(t)S(t)I(t)}{N} - (\gamma(t) + \mu(t))I(t) \\
\dot{R}(t) &= \gamma(t)I(t) \\
\dot{D}(t) &= \mu(t)I(t),
\end{aligned}
\tag{1}
$$

where $S(t)$, $I(t)$, $R(t)$, and $D(t)$ are the numbers of susceptible, infected, recovered, and deceased individuals, respectively, and N is the total population, $N = S(t) +$

$I(t) + R(t) + D(t)$. Without loss of generality, N is a large number and difficult to compute in some optimization problems. Thus, we used the fraction of the population in each, which means that $S(t)$, $I(t)$, $R(t)$, and $D(t)$ are divided by N. In this way, our calculation is simpler, and we can still adhere to the same dynamic system of the epidemic.

The parameter $\beta(t)$ represents the number of contacts each day for infected individuals. Besides, there are two essential assumptions. First, contacts between the infected and uninfected people are sufficient to spread the disease. Second, the population is mixed homogeneously. Thereby, $\beta(t)S(t)$ susceptible people are infected by a patient each day on average, and $\beta(t)S(t)I(t)$ is the number of newly infected people, $\gamma(t)$ is the rate at which infected cases become recovered. Then, there are $\gamma I(t)$ infected individuals turned into the third compartment. Similarly, $\mu(t)$ is the rate of mortality by disease, and $\mu(t)I(t)$ infected individuals are deceased.

2.2. Reproduction Number

There is an essential threshold used to get a better understanding of the transmission rate. This threshold is characterized by the reproduction number, which is an estimate of newly infected cases caused by an infected individual [25]. Assuming there is just one person infected in the beginning, then in the SIRD model, $I_0 = 1$, $S_0 = N - 1$, and no recovery at this moment. When we utilized the fraction for each compartment, $I_0 \approx 0$ and $S_0 \approx 1$. By the second equation of eq. (1), we have:

$$\dot{I}(t) = I(t)[\beta(t)S(t) - (\gamma(t) + \mu(t))], \tag{2}$$

where $\beta(t), \gamma(t),$ and $\mu(t)$ are greater than 0. In eq. (2), it is simple to obtain $\dot{I} > 0$ when $\beta(t)S(t) > (\gamma(t) + \mu(t))$, which means the number of infected cases keeps increasing.

A key idea about the reproduction number R_0 is that it is the gauge of the secondary infections at the beginning of this disease invasion [9]. After the disease invasion, the contact rate β decreases because of the reduction of this susceptible group. Those who are infected or have already recovered cannot be infected once again. Thereby, another similar threshold, which is the effective reproduction number R_e [22], is used to indicate whether the disease keeps spreading, and R_0 is the upper bound of R_e [12].

$$R_e(t) = \frac{\beta(t)S(t)}{\gamma(t) + \mu(t)} < \frac{\beta(0)}{\gamma(0) + \mu(0)} = R_0. \tag{3}$$

3. Deep Neural Network

In this section, we provide a brief description of the deep learning neural network architectures utilized in our algorithm for parameter identification and disease dynamics prediction.

3.1. Feedforward Neural Network

The feedforward neural network (FNN) is an artificial neural network with a simple architecture in which the connections between the nodes do not form a cycle. FNN is

utilized to approximate the target function by applying several activation functions to input recursively and then output a value or vector that is close to the target values.

The input layer consists of input neurons, bringing the training data into the network for further processing by hidden layers. Layers between the input layer and output layer are hidden layers, which perform linear or nonlinear transformations of training data to produce an intended output. These transformations are implemented by activation functions. The first hidden layer takes the weighted sums of input through an activation function to produce output for each neuron. Then, these outputs plus a constant, which is bias in the first hidden layer, are viewed as an input of the second hidden layer. After that, as the last layer of the network, the output layer presents desired results [6].

The forward propagation from one layer to the next layer's nodes is given as follows [7].

$$z_j^{l+1} = \sum_i^{n_l} w_{ij}^l f_{l-1}(z_i^l) + b^l, \tag{4}$$

where n_l represents the number of neurons in the lth layer, z_i^l, $i = 1, \cdots, n_l$, denotes the output of the ith node in $(l-1)^{th}$ layer, f represents the activation function, w_{ij}^l are the weights between node i^{th} and node j^{th}, b^l are the bias in the lth layer. The activation functions employed in this study are the hyperbolic tangent function

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}, \tag{5}$$

and sigmoid function

$$\sigma(x) = \frac{1}{1 + e^{-x}}. \tag{6}$$

When output values are obtained from the output layer, they are used to build a loss function, which is used to estimate the error of the model. An optimizer, like Adam [14] or gradient descent method [1], is employed to minimize the loss function so that the output is close to the observations. According to different problems, the selections of optimizer and loss function are different.

3.2. Physics-Informed Neural Network

The physics-inform neural network (PINN) is a data-driven algorithm to approximate the solution of differential equations and identify parameters [20]. It could utilize any type of neural network architectures, like the FNN and Convolutional neural networks (CNN), as the main framework. The applied activation functions and optimization methods in PINN are the same as the usual deep learning techniques. The fascinating part of this algorithm is the loss function, which is generally comprised of boundary conditions, initial values, and physical constraints.

The outputs of the neural network are constrained to satisfy the system of differential equations by penalizing the residuals of differential equations into the loss function. In these residual equations, the derivatives of outputs with respect to time

are computed by a black-box, which is automatic differentiation [17]. Automatic differentiation is employed to train these models and allows PINN to take derivatives with respect to input coordinates so that it could discover the latent physical laws. More details could be found in [20].

3.3. Long Short-term Memory Neural Network

As an advanced tool to process and predict time-series, LSTM is a special type of RNN. The difference between RNN and feedforward neural network is that it is capable of storing past information by taking previous layers' output as an input for the next input. Also, all layers of RNN share the same parameters because they perform the same tasks for the elements of the sequence, which is a so-called recurrent. Based on this feature, RNN could predict sequential information. However, on account of only involving the output of the last layer, it cannot work out long-term memory problems [10]. Thus, LSTMs are created for solving them.
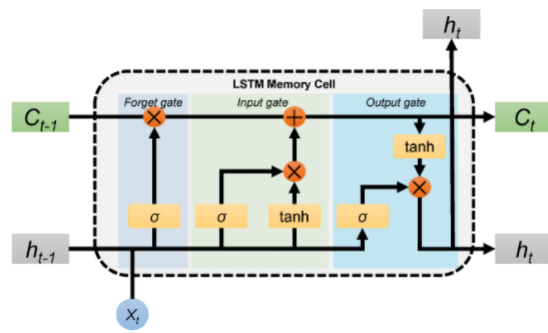

Figure 2. Architecture of LSTM Cell [4]

LSTMs as shown in Figure 2 are chains of memory cells containing three gates, which are input gate, forget gate, and output gate as explained in [4]. The key part of these memory cells is the cell state that could maintain the global information of sequence in each time-step. The content of the cell state would be modified at different time-steps through the forget gates and input gates. More specifically, forget gates decide what information would be thrown away by utilizing the sigmoid function. Instead, input gates decide what new information would be stored by combining sigmoid eq. (6) and tanh eq. (5) functions. Then, the cell state is updated and moves forward to the next state by implementing some operations. After that, the output is calculated in the output gate. By analyzing the structure of a memory cell, we could see that LSTMs are capable of storing long-term memory by putting new information into cell states.

4. Data and Algorithms

In this section, we describe the collected data and introduce our algorithms for parameter identification and making predictions. In our work, the parameters of the SIRD model are viewed as daily-varying and weekly-varying parameters to simulate the complex real situation of COVID-19. Based on this perspective, two algorithms are employed to identify daily and weekly time-series parameters. Having the identified parameters by PINN as an input, we introduce the LSTM algorithm for making predictions.

4.1. Data

The data considered in our study is downloaded from `https://covidtracking.com/data/download` for New York, New Mexico, and Texas states. It is comprised of commutative infected, recovered, and deceased cases for the period from March 30 to September 30. A plot of the data is shown in Figure 3.
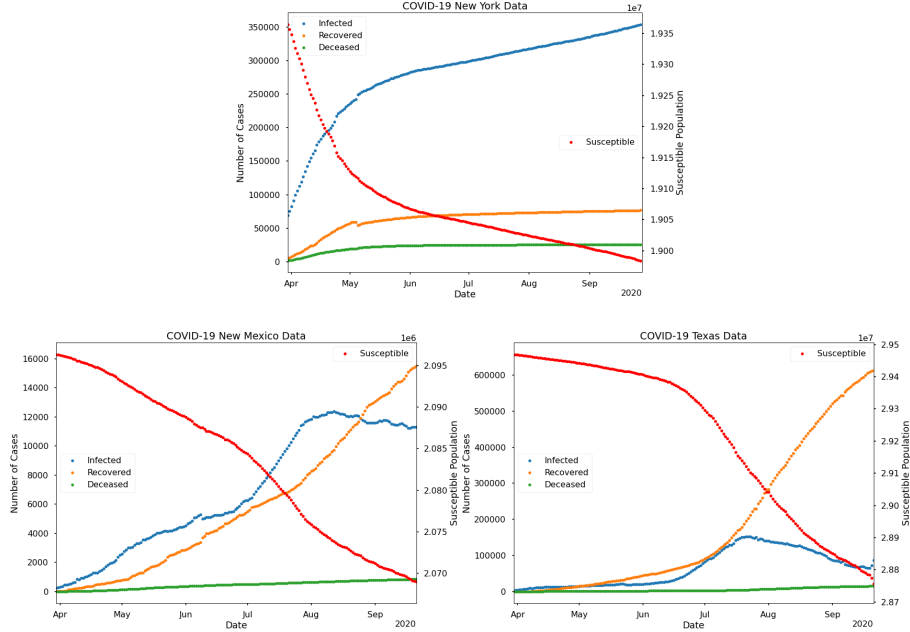

**Figure 3.** COVID-19 data for New York, New Mexico, and Texas

As can be observed, the three states have different dynamics between the compartments. In New York, while the number of cases in I, R, and D compartments is increasing, the number of recovered and deceased cases is relatively small compared to infections. New Mexico data shows that the infected and recovered cases kept almost the same increasing rate before August but then the infections started decreasing slowly, while the number of recovered individuals kept increasing at a high rate. As for Texas, the infected population started growing fast from the second half of June, and then went down after about 30 days.

4.2. Parameter Identification Algorithm

In this subsection, we describe how to identify parameters of the SIRD model by utilizing PINN.

4.2.1. PINN Architecture for SIRD Model

The utilized PINN for learning the parameters is composed of an FNN architecture. This architecture consists of 4 hidden layers and 60 neurons in each layer. The activation functions of hidden layers and the output layer are the hyperbolic tangent function, and sigmoid function, respectively.
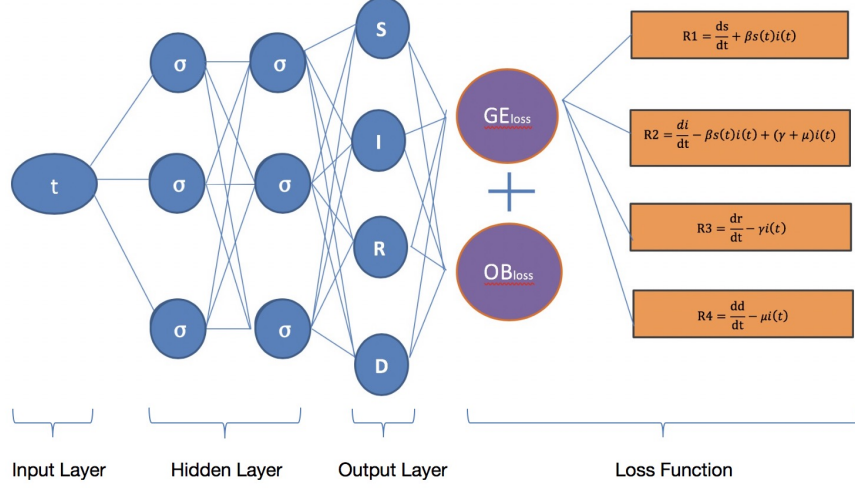
Figure 4. Physics Informed Neural Network

Figure 4 shows the basic framework, which is FNN, and the loss function of PINN in our study. The loss function of PINN has two parts, namely GE_{loss} and OB_{loss} in this figure. The GE_{loss} represents the residuals that are obtained from the SIRD model eq. (1) by subtracting the right side from the left side:

$$
\begin{aligned}
R_1 &= \frac{dS}{dt} + \beta(t)S(t)I(t) \\
R_2 &= \frac{dI}{dt} - \beta(t)S(t)I(t) + (\gamma(t) + \mu(t))I(t) \\
R_3 &= \frac{dR}{dt} - \gamma(t)I(t) \\
R_4 &= \frac{dD}{dt} - \mu(t)I(t).
\end{aligned}
\tag{7}
$$

The OB_{loss} is the mean squared error between the outputs of the neural network and data. We employ Adam algorithm, which is the first-order gradient-based optimization of stochastic objective functions [14], to update the parameters of the neural network by minimizing the loss function.

4.2.2. Parameters Identification Algorithm

Algorithm 1 describes how to use PINN to identify the daily time-varying parameters. The input is time t, and outputs include the three parameters and the four

compartments of the SIRD model. The weights w and bias b are initialized randomly.

Algorithm 1: PINN identify daily time-varying parameters

Data: S,I,R,D,t

Initialize weights w, bias b;

for *number of epochs* **do**

Each compartment of SIRD is obtained from forwarding propagation of neural network

$$\hat{S}, \hat{I}, \hat{R}, \hat{D}, \beta, \gamma, \mu = NN(t)$$

Build the loss function components, where N_{ob} is the number of collected data of each compartment, and N_f is the number of collocation points

$$OB_{loss} = \frac{1}{N_{ob}} \sum_{i=1}^{N_{ob}} \left(\left| \hat{S}^i - S^i \right| + \left| \hat{I}^i - I^i \right| + \left| \hat{R}^i - R^i \right| + \left| \hat{D}^i - D^i \right| \right)$$

OB_{loss} denotes the difference between the output of the neural network and observation data

$$GE_{loss} = \frac{1}{N_t} \sum_{i=1}^{N_t} \left(\left| \frac{d\hat{S}^i}{dt^i} + \beta \hat{S}^i \hat{I}^i \right| + \left| \frac{d\hat{I}^i}{dt^i} - \beta \hat{S}^i \hat{I}^i + (\gamma + \mu \hat{I}^i) \right| \right.$$

$$\left. + \left| \frac{d\hat{R}^i}{dt^i} - \gamma \hat{I}^i \right| + \left| \frac{d\hat{D}^i}{dt^i} - \mu \hat{D}^i \right| \right)$$

GE_{loss} represents the SIRD model residual loss, then we have the total loss

$$loss = OB_{loss} + GE_{loss}$$

Using the Adam optimizer to update the weights and bias by minimizing the loss function

end

The approach of identifying the weekly-varying parameters is described in algorithm 2. We divide the utilized data of algorithm 1 into weekly intervals, and then they are trained to identify corresponding parameters. Cubic spline interpolation is employed

to obtain adequate training set for the neural network training.

Algorithm 2: PINN identify weekly time-varying parameters

Data: S,I,R,D,t

Form the weekly data;

for *each week* **do**

    $S, I, R, D, t = CubicSpline(S, I, R, D, t)$;

    $N = length(t)$;   ▷`Each dataset has the same length`

    Initialize weights w, bias b,β,γ,μ;

    for *number of epochs* **do**

        Each compartment of SIRD is obtained from forwarding propagation of neural network

$$\hat{S}, \hat{I}, \hat{R}, \hat{D} = NN(t)$$

        Build the loss function components.

$$OB_{loss} = \frac{1}{N}\sum_{i=1}^{N}\left(\left|\hat{S}^i - S^i\right|^2 + \left|\hat{I}^i - I^i\right|^2 + \left|\hat{R}^i - R^i\right|^2 + \left|\hat{D}^i - D^i\right|^2\right)$$

        OB_{loss} denotes the difference between the output of the neural network and observation data

$$GE_{loss} = \frac{1}{N}\sum_{i=1}^{N}\left(\left|\frac{d\hat{S}^i}{dt^i} + \beta\hat{S}^i\hat{I}^i\right|^2 + \left|\frac{d\hat{I}^i}{dt^i} - \beta\hat{S}^i\hat{I}^i + (\gamma + \mu\hat{I}^i)\right|^2 \right.$$
$$\left. + \left|\frac{d\hat{R}^i}{dt^i} - \gamma\hat{I}^i\right|^2 + \left|\frac{d\hat{D}^i}{dt^i} - \mu\hat{D}^i\right|^2\right)$$

        GE_{loss} represents the SIRD model residual loss

$$loss = OB_{loss} + GE_{loss}$$

        Using the Adam optimizer to update the weights and bias by minimizing the loss function

    end

    save the value of β γ and μ

end

4.3. LSTM for Prediction of Infectious Cases

Having the learned parameters by PINN, the LSTM is employed to predict the future parameters by applying Keras as depicted in Figure 5. At first, we normalize these parameters and create new datasets with multiple inputs and one output. Inputs are previous three-time steps parameters, and output, which is the prediction of these inputs, is the next one-time step parameter. Then, the data is used to train the LSTM neural network. We utilize three hidden layers of 80 nodes each to predict the parameters of the next four weeks.
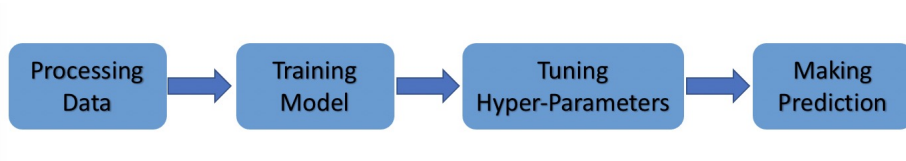
Figure 5. Flow-Chart of LSTM

5. Simulation

In this section, we present and discuss our simulation results. The accuracy of the learned parameters by PINN is validated using the relative error in the numerical solution of System (1), obtained using the Fourth Order Runge-Kutta (RK4) method, with respect to the exact solution. Furthermore, the relative error in the learned solution by PINN with respect to the data is examined. We start by presenting the results for the learned daily and weekly time-varying parameters followed by the associated reproduction numbers. Then, predictions of infectious cases are provided.

5.1. Time-Varying Parameters

In general, the daily-varying values of the parameters are more reflective of the daily data than the single constant average values of these parameters. However, identifying the daily-varying parameters is traditionally costly and adds to the complexity of the problem. Fortunately, PINN provides an efficient approach to overcome this difficulty. On the other hand, our simulations show that the learned weekly-varying data produce precisely identified parameters and more stable approximation results.

Algorithms 1 and 2 give the identified parameters and the learned values of the SIRD model by PINN. We use the RK4 method to solve the SIRD model with learned parameters. Then, the solutions of the ODE system (1) are compared with the learned values.

In Figures 6, 7, and 8, we present the learned values by PINN (NN), the solution of the ODE system (1) (ode), and the reported data (data). Moreover, the relative errors with respect to the reported data are also computed.
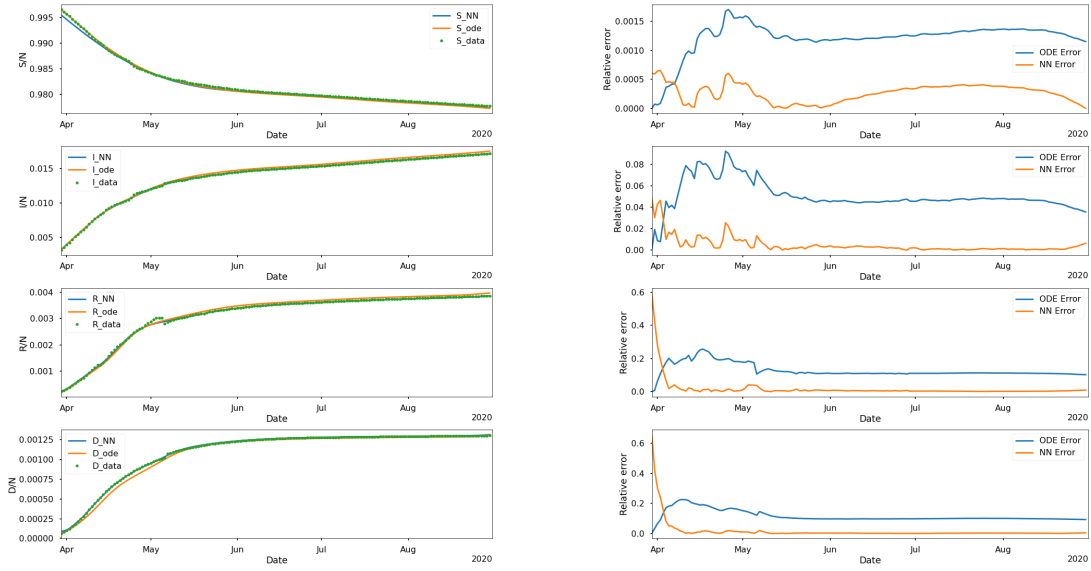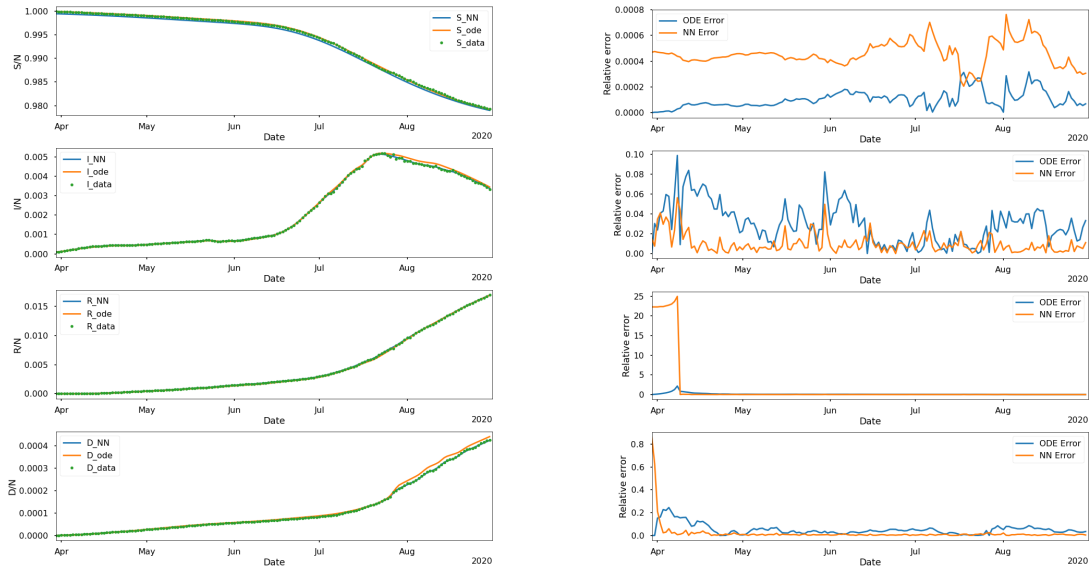
Figure 6. Simulation Results of New York

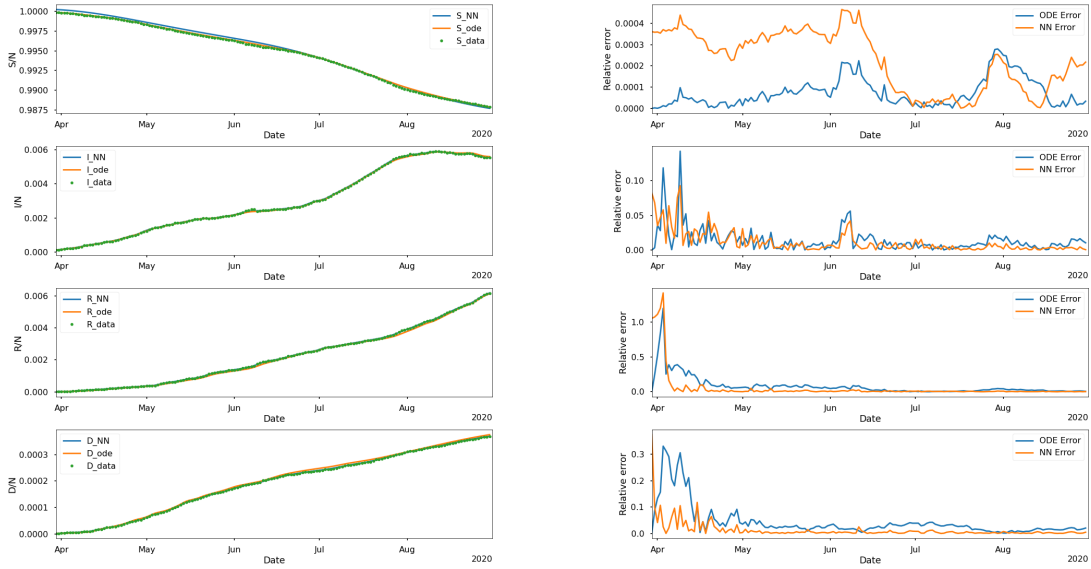**Figure 7.** Simulation Results of Texas

12

Figure 8. Simulation Results of New Mexico

There are some large relative errors at the early stage of the outbreak of the disease. This could be due to the small size of the recovered and deceased populations. But the relative errors of the solution of the SIRD model are in a reasonable range, which means that we obtained ideal identified parameters. For the weekly-varying parameters, the period is divided into weeks and the learned results are presented in Figures 9, 10, and 11.
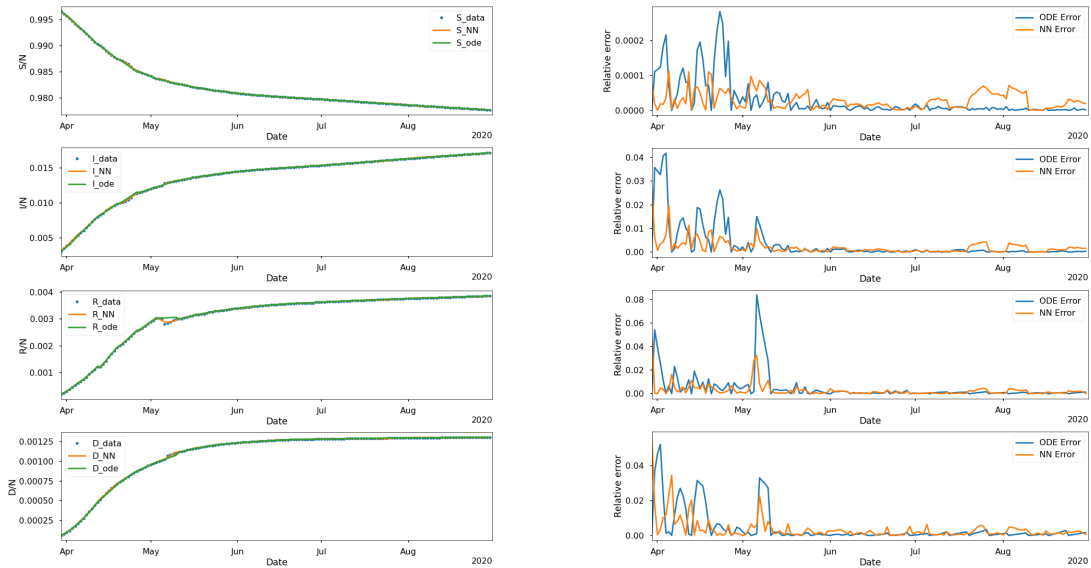

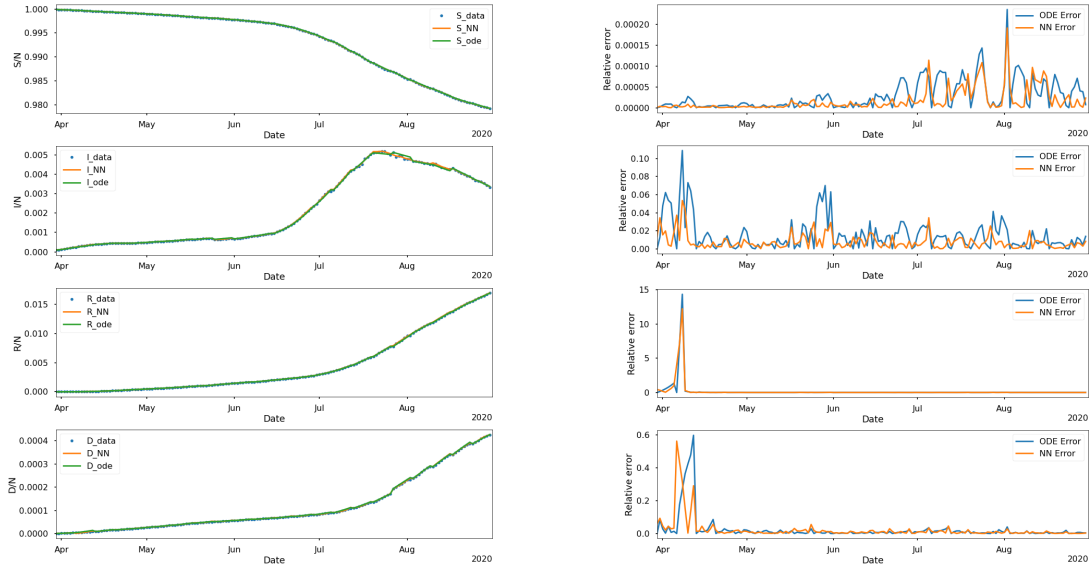**Figure 9.** Simulation Results of New York
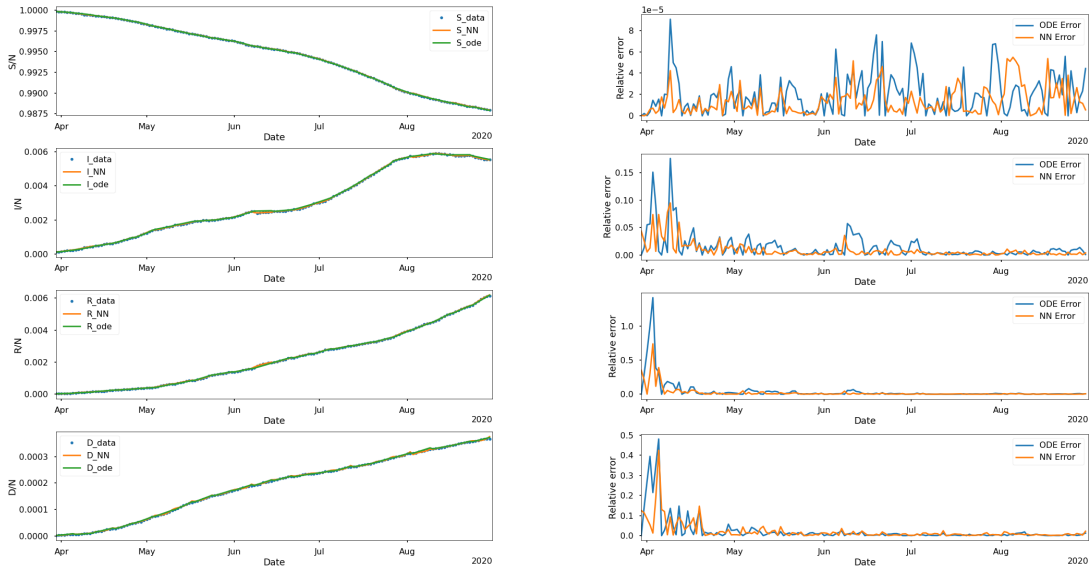
Figure 10. Simulation Results of Texas

Figure 11. Simulation Results of New Mexico

The relative errors in Figures 9, 10, and 11 are smaller compared to the results in Figures 6, 7, and 8. Thus, we could infer that the approximation of constant parameters is more accurate and stable.

5.2. *Simulated Reproduction Numbers*

By eq. (3), the effective reproduction number can be obtained using the learned daily and weekly varying parameters. Figure 12 shows the effective reproduction number in

14

the three states. We could see the daily and weekly time-varying reproduction numbers have the similar trends, which proves the feasibility and effectiveness of our methods to some extent. In the result of New York, we could see the R_e is greater than 1 in the whole period, which represents that the number of infectious cases keeps increasing. As for the situation in New Mexico, R_e decreases to 1 from August, then the population of infected individuals decreased at that period. By observing the change of R_e in Texas, the value is too large in the beginning because the parameter identification in that period is not ideal. But after that period, the R_e goes back to normal and oscillates between 0 and 4. Besides, we could see that the growth rate of infected individuals in New York is faster than the other two states by Figure 3.
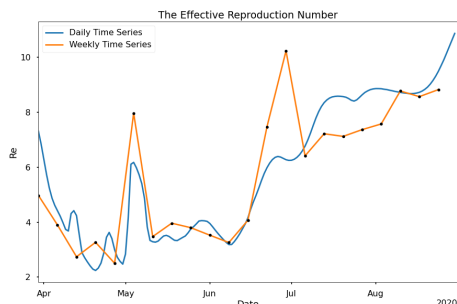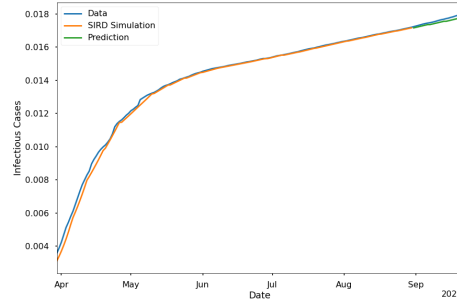

(a) New York
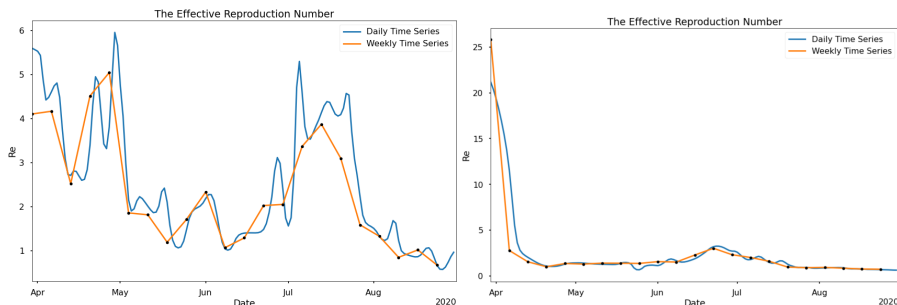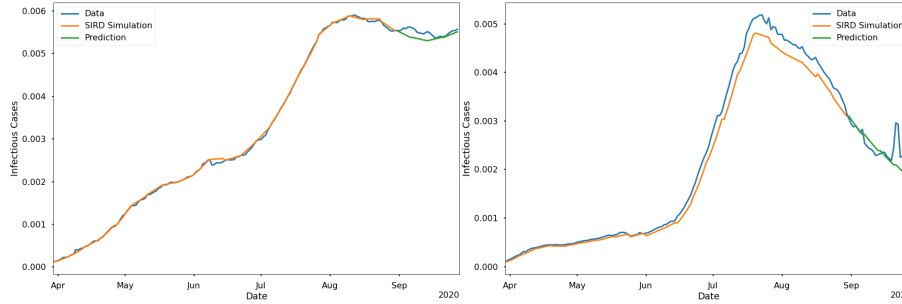

(b) New Mexico

(c) Texas

Figure 12. Effective Reproduction Number

5.3. Prediction of Infectious Cases

The LSTM is employed to predict the future values of the parameters. Figure 13 presents the predictions of infected cases for the next four weeks. The prediction of New York is close to the original trend of infectious cases. Predictions in New Mexico show general accord with real situations, but it cannot capture the small fluctuations. As for Texas, there is a sharp increase and decrease in the infectious cases at the end of September, and our prediction could not detect this situation.

(a) New York

(b) New Mexico

(c) Texas

Figure 13. Predictions

6. Conclusions

We introduced a data-driven deep learning approach based on physics informed neural network to solve the epidemiological models and identify weekly and daily time-varying parameters. The PINN was used for parameters identification and solving the ODE system. LSTM was implemented to predict the weekly time-varying parameters and then substituted predicted parameters into the SIRD model so that we could obtain the future trend of COVID-19. The results and errors have shown that the weekly time-varying parameters provided a good fit to the real data. The algorithms could be further developed to achieve a more accurate approximation for complex problems. We intend to explore other architectures of PINN in future work.

References

[1] S.i. Amari, *Backpropagation and stochastic gradient descent method*, Neurocomputing 5 (1993), pp. 185–196.

[2] J.M. Carcione, J.E. Santos, C. Bagaini, and J. Ba, *A simulation of a COVID-19 epidemic based on a deterministic SEIR model*, arXiv:2004.03575 (2020).

[3] Y.C. Chen, P.E. Lu, C.S. Chang, and T.H. Liu, *A Time-dependent SIR model for COVID-19 with undetectable infected persons*, IEEE Transactions on Network Science and Engineering 7 (2020), pp. 3279–3294.

[4] H. Fan, M. Jiang, L. Xu, H. Zhu, J. Cheng, and J. Jiang, *Comparison of Long Short Term Memory Networks and the Hydrological Model in Runoff Simulation*, Water 12 (2020), p. 175.

[5] L. Ferrari, G. Gerardi, G. Manzi, A. Micheletti, F. Nicolussi, and S. Salini, *Modelling provincial COVID-19 epidemic data in Italy using an adjusted time-dependent SIRD model*, arXiv:2005.12170 (2020).

[6] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016, `http://www.deeplearningbook.org`.

[7] D.L. Gray and A.N. Michel, *A training algorithm for binary feedforward neural networks*, IEEE Transactions on Neural Networks 3 (1992), pp. 176–194.

[8] D. Guanghong, L. Chang, G. Jianqiu, W. Ling, C. Ke, and Z. Di, *SARS epidemical forecast research in mathematical model*, Chinese Science Bulletin 49 (2004), pp. 2332–2338.

[9] R.J. Hatchett, C.E. Mecher, and M. Lipsitch, *Public health interventions and epidemic intensity during the 1918 influenza pandemic*, Proceedings of the National Academy of Sciences 104 (2007), pp. 7582–7587.

[10] S. Hochreiter and J. Schmidhuber, *Long short-term memory*, Neural Computation 9 (1997), pp. 1735–1780.

[11] H. Jo, H. Son, S.Y. Jung, and H.J. Hwang, *Analysis of COVID-19 spread in South Korea using the SIR model with time-dependent parameters and deep learning*, medRxiv (2020).

[12] A. Kergassner, C. Burkhardt, D. Lippold, S. Nistler, M. Kergassner, P. Steinmann, D. Budday, and S. Budday, *Meso-scale modeling of COVID-19 spatio-temporal outbreak dynamics in Germany*, medRxiv (2020).

[13] W.O. Kermack and A.G. McKendrick, *A contribution to the mathematical theory of epidemics*, Proceedings of the Royal Society of London. Series A, Containing papers of a mathematical and physical character 115 (1927), pp. 700–721.

[14] D.P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, arXiv:1412.6980v9 (2017).

[15] Q. Lin, S. Zhao, D. Gao, Y. Lou, S. Yang, S.S. Musa, M.H. Wang, Y. Cai, W. Wang, L. Yang, *et al.*, *A conceptual model for the outbreak of Coronavirus disease 2019 (COVID-19) in Wuhan, China with individual reaction and governmental action*, International Journal of Infectious Diseases 93 (2020), pp. 211–216.

[16] Z. Liu, P. Magal, O. Seydi, and G. Webb, *Predicting the cumulative number of cases for the COVID-19 epidemic in China from early data*, Mathematical Biosciences and Engineering 17 (2020), pp. 3040–3051.

[17] L. Lu, X. Meng, Z. Mao, and G.E. Karniadakis, *DeepXDE: A deep learning library for solving differential equations*, arXiv:1907.04502v2 (2020).

[18] L. Magri and N.A.K. Doan, *First-principles machine learning modelling of COVID-19*, arXiv:2004.09478 (2020).

[19] M. Rafiq, J. Macías-Díaz, A. Raza, and N. Ahmed, *Design of a nonlinear model for the propagation of COVID-19 and its efficient nonstandard computational implementation*, Applied Mathematical Modelling 89 (2021), pp. 1835–1846.

[20] M. Raissi, P. Perdikaris, and G.E. Karniadakis, *Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations*, Journal of Computational Physics 378 (2019), pp. 686–707.

[21] M. Raissi, N. Ramezani, and P. Seshaiyer, *On parameter estimation approaches for predicting disease transmission through optimization, deep learning and statistical inference methods*, Letters in Biomathematics 6 (2019), pp. 1–26.

[22] B. Ridenhour, J.M. Kowalik, and D.K. Shay, *Unraveling R_0: Considerations for Public Health Applications*, American Journal of Public Health 108 (2018), pp. S445–S454.

[23] F. Shahid, A. Zameer, and M. Muneeb, *Predictions for COVID-19 with deep learning models of LSTM, GRU and Bi-LSTM*, Chaos, Solitons & Fractals 140 (2020), p. 110212.

[24] Y. Shi, *Stochastic dynamic model of SARS spreading*, Chinese Science Bulletin 48 (2003), pp. 1287–1292.

[25] P. van den Driessche, *Reproduction numbers of infectious disease models*, Infectious Disease Modelling 2 (2017), pp. 288–303.

[26] H. Wang, Z. Wang, Y. Dong, R. Chang, C. Xu, X. Yu, S. Zhang, L. Tsamlag, M. Shang, J. Huang, *et al.*, *Phase-adjusted estimation of the number of coronavirus disease 2019 cases in Wuhan, China*, Cell Discovery 6 (2020), pp. 1–8.

[27] J. Wang, A.J. McMichael, B. Meng, N.G. Becker, W. Han, K. Glass, J. Wu, X. Liu, J. Liu, X. Li, *et al.*, *Spatial dynamics of an epidemic of severe acute respiratory syndrome in an urban area*, Bulletin of the World Health Organization 84 (2006), pp. 965–968.

[28] World Health Organizaiton, *Rolling updates on coronavirus disease (COVID-19)* (2020). Available at `https://www.who.int/emergencies/diseases/novel-coronavirus-2019/events-as-they-happen`, Updated 31 July 2020.

[29] Worldometers, *COVID-19 CORONAVIRUS PANDEMIC* (2020). Available at `https://www.worldometers.info/coronavirus/?#countries`.

[30] Z. Yang, Z. Zeng, K. Wang, S.S. Wong, W. Liang, M. Zanin, P. Liu, X. Cao, Z. Gao, Z. Mai, *et al.*, *Modified SEIR and AI prediction of the epidemics trend of COVID-19 in China under public health interventions*, Journal of Thoracic Disease 12 (2020), p. 165.

[31] A. Zeroual, F. Harrou, A. Dairi, and Y. Sun, *Deep learning methods for forecasting covid-19 time-series data: A comparative study*, Chaos, Solitons & Fractals 140 (2020), p. 110121.