



Differential Neural Networks Prediction Using Slow and Fast Hybrid Learning: Application to Prognosis of Infections and Deaths of COVID-19 Dynamics

A. Poznyak¹ · I. Chairez² · A. Anyutin³

Accepted: 26 February 2023 / Published online: 23 March 2023

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2023

Abstract

This essay discusses a potential method for predicting the behavior of various physical processes and uses the COVID-19 outbreak to demonstrate its applicability. This study assumes that the current data set reflects the output of a dynamic system that is governed by a nonlinear ordinary differential equation. This dynamic system may be described by a Differential Neural Network (DNN) with time-varying weights matrix parameters. A new hybrid learning scheme based on the decomposition of the signal to be predicted. The decomposition considers the slow and fast components of the signal which is more natural to signals such as the ones corresponding to the number of infected and deceased patients who suffered of COVID 2019 sickness. The paper results demonstrate the recommended method offers competitive performance (70 days of COVID prediction) in comparison to similar studies.

Keywords Artificial neural networks · Hybrid learning · Virus evolution · COVID

1 Introduction

1.1 Predictions Based on Neural Networks

This section reviews how neural networks may predict the evolution of signal temporal evolution.

✉ A. Poznyak
apoznyak@ctrl.cinvestav.mx

I. Chairez
isaac.chairez@tec.mx

¹ CINVESTAV IPN, DCA, Cd. de Mexico, Mexico

² Tecnológico de Monterrey, Institute of Advanced Materials for Sustainable Manufacturing, Cd. de Guadalajara, Mexico

³ Institute of Radio Engineering and Electronics, Fryazino Branch, Ran, Fryazino, Russia

1.1.1 Prediction by Static (Feed-Forward) NN

A set of input values are considered by each neuron. Each one is associated with a *weight*, which is a varying value that may be determined by supervised or unsupervised training techniques like data *clustering*, and a *bias*. The network selects a neuron's output depending on its weight and bias. All such activities in the context of *Classification* require labeled datasets. You thus require guided learning. In supervised learning, people verify that the neural network's predictions are accurate. This aids the neural network in comprehending how labels and data related. Face identification, picture recognition and labeling, voice detection, and speech transcription are a few examples of this. Deep learning can link pixels in a picture and a person's name via categorization. The act of grouping or *clustering* is the identification of commonalities. Understand that labels are not always necessary for the deep learning model to detect commonalities. Unsupervised learning is when a system utilizes machine learning to learn on its own when there are no helping human labels from which to draw. This keeps the possibility of creating extremely precise models. Customer churn is a type of clustering.

As we are all aware, predictive analytic uses methods like predictive modeling and machine learning to examine historical data and forecast future patterns [7]. Contrary to conventional forecasting techniques, neural networks are unique. In contrast to a neural network, the most popular model, linear regression, is actually a pretty straightforward approach to problem-solving. Because of their hidden layers, neural networks do predictive analytic more effectively. Only input and output nodes are used in linear regression models to generate predictions. The hidden layer is also used by the neural network to improve prediction accuracy. That's because it *learns* similarly to how people do. So why isn't neural network prediction used by everyone? They are prohibitively expensive due to their high computer power requirements. In addition, massive data sets are required to train neural networks, which your company might not have. But as IT technology becomes more affordable, the first obstacle could soon vanish. Soon, there won't be any more "unpleasant shocks" because to technologies like *Artificial Neural Networks* (ANNs).

1.1.2 Prediction by Dynamic (with Feedback) NN

ANNs are often regarded as effective instruments for modeling intricate, nonlinear systems using hazy dynamic models. ANNs were first utilized as reliable predictors of various processes with static reliance on input–output data. The time effect should be included in the ANN when it must be used to describe a rough model of time-dependent input–output interactions, which necessitates the creation of a dynamic ANN or DNN [11]. In continuous time modelling we will be referred to DNN as *Differential Neural Networks*. The review [13] lays forth the different recurrent and differential forms of *Dynamic Neural Networks* (DNN), their mathematical construction, and techniques for adjusting the network weights. The characteristics of DNNs motivate their use to represent the dynamics of decontamination processes. This review details recent findings on the DNN application for the modelling and controlling of treatment systems based on either biological or chemical processes. The modeling application of DNN for common methods used in the treatment of wastewater, contaminated soil, and the atmosphere is described. The major benefits of using the approximate DNN-based model instead of designing the complex mathematical description for each treatment are analyzed to enhance the efficiency of the decontamination treatment. In this paper, we also highlight the remarkable efficiency of DNNs as a keystone tool for *modelling of epidemics*. [15, 18].

1.2 On Mathematical Predictions of Epidemics

In the last few years, researchers and government officials have used computer-based models to try to forecast the course of the coronavirus pandemic (see, for example, [2, 8, 10, 19]). To predict the future of the coronavirus disease 2019 (COVID-19) outbreaks globally, several mathematical models have been developed. These forecasts have a significant impact on how soon and forcefully, governments respond to an outbreak. However, rather than producing accurate quantitative predictions regarding the magnitude or duration of illness burdens, the primary and most efficient application of epidemiological models to evaluate the relative efficacy of different treatments in lowering disease burden.

There are several studies remarking that models are hardly crystal balls when it comes to making predictions, and according to science journalist Miles O'Brien (PBS News Hour), "all of them require human assumptions" [1]. The creation of these models and their eventual goal are more sophisticated than many of us think, according to specific research periodicals. Our world is complex and has more data than knowledge. The Global Epidemic and Mobility Model, or GLEAM, is curated by a group of bio-statisticians at Seattle's Fred Hutchinson Cancer Research Center [3]. They create mathematical models that explain how infections spread chaotically and exponentially. According to the projection from last month, 17,000 to 29,300 additional fatalities would likely be reported in the US solely for the week ending February 13, 2021, totaling 465,000 to 508,000 COVID-19 deaths by this time. The accuracy of mathematical forecasts in battling epidemics is still being worked on. Nevertheless, creating such illness prediction models is a crucial issue for scientific societies worldwide. It necessitates prompt and comprehensive answers, including a potential application for defining new politics and prevention schemes.

1.3 Main Concepts of This Paper

The results presented here are based on three principle concepts:

- Although we have hundreds of years of theoretical knowledge on how to create mathematical models of infectious diseases, have any of these models ever been put to the test using all of the data sources at our disposal in real-time? No. As we create this automobile and learn more about these models, it is hurtling down the highway. For a more accurate model design, **it is really difficult to take into consideration all human aspects (social, informational, climatic, and others) acting during sickness.**
- Any recommended model must include the inherent uncertainties associated with the most recent data. Thus, for instance, we lack sufficient statistical data to accept all of the conditions that should be satisfied to use any stochastic prediction models that are accessible (such as the Kalman filter or any of its modifications such as a requirement for noises to have Gaussian distributions with known covariation matrices, local linearity of the model, exact knowledge all participating parameters and so on). We only have one data trajectory (realization), making it complicated to apply statistical concepts like mathematical expectation (mean value), variance, and confidence interval. We can also not repeat the experiment to get at least one other data curve. **This indicates that a statistical method for this kind of problem is not applicable!**
- Given the previous items, we suppose that **the current data-set represents the output of some dynamic system governed by a nonlinear ordinary differential equation** and may be modeled by a Differential Neural Network with time-varying weights matrix

parameters whose dynamics is governed by special *Learning laws* containing *slow* and *fast* components.

All results reported below justify nice performances of the suggested approach.

2 DNN Model with Slow and Fast Learning

2.1 Ideas of a Prediction Algorithm for Models with Complete Information

2.1.1 Non-causal Model

Consider initially an ideal scenario where we know that the following mathematical model produces the scalar output $x(t) \in \mathbb{R}$ of any dynamic plant.

$$\left. \begin{aligned} x^{(n)}(t) &= f(t, x(t), \dot{x}(t), \dots, x^{(n-1)}(t)) \\ x^{(r)}(0) &= x_0^{(r)}, r = 0, 1, \dots, n-1 \end{aligned} \right\} \tag{1}$$

where the nonlinear function $f : \mathbb{R}_+ \times \mathbb{R}^n \rightarrow \mathbb{R}$ and initial condition x_0 supposed to be known exactly. Defining vector $\mathbf{x}(t) \in \mathbb{R}^n$ with components

$$x_1(t) := x(t), x_2(t) := \dot{x}_1, \dots, \dot{x}_{n-2} := x_{n-1}, x_n := \dot{x}_{n-1}, \tag{2}$$

we can represent (1) as

$$\left. \begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{F}(t, \mathbf{x}(t)) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}v(t) \\ \mathbf{A} &= \begin{pmatrix} 0 & 1 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 & 0 \\ \vdots & 0 & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & \dots \\ 0 & 0 & \dots & 0 & 0 & 0 & 0 \end{pmatrix} \in \mathbb{R}^{n \times n}, \mathbf{b} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} \in \mathbb{R}^{n \times 1} \\ v(t) &= f(t, x_1(t), x_2(t), \dots, x_{n-1}(t)) \in \mathbb{R} \end{aligned} \right\} \tag{3}$$

In the corresponding integral form the differential model (3) can be rewritten as

$$\begin{aligned} \mathbf{x}(t+T) &= \mathbf{x}(t) + \int_{\tau=t}^{t+T} \mathbf{F}(\tau, \mathbf{x}(\tau)) d\tau = \mathbf{x}(t) + T \mathbf{r}(t, T) \\ \mathbf{r}(t, T) &:= \frac{1}{T} \int_{\tau=t}^{t+T} \mathbf{F}(\tau, \mathbf{x}(\tau)) d\tau \end{aligned} \tag{4}$$

where the variable $\mathbf{r}(t, T)$ represents the "averaged rate" of changing the considered output variable $\mathbf{x}(t)$ on the time-interval $[t, t+T]$. Considering the data set $\{\mathbf{x}(\tau)\}_{\tau \in [0, t]}$ as the information on the process available up to the moment t we may conclude that $\mathbf{r}(t, T)$ (4) contains the information on nearest future $\{\mathbf{x}(\tau)\}_{\tau \in [t, t+T]}$ with the horizon T and hence may be considered as "non-causal".

2.1.2 Causal Approximation

Introduce standard operators of delay e^{-sT} and differentiation s acting as

$$e^{-sT} f(t) = f(t - T), sf(t) = f'(t)$$

and using the local approximation

$$e^{sT} \simeq 1 + sT + \frac{T^2}{2}s^2 + \frac{T^3}{6}s^3$$

for the "forecasting operator" e^{sT} , we can obtain the following approximate relation:

$$\left. \begin{aligned} \mathbf{r}(t, T) &= e^{sT} e^{-sT} \mathbf{r}(t, T) = e^{sT} \mathbf{r}(t - T, T) \simeq \\ &\left(1 + sT + \frac{T^2}{2}s^2 + \frac{T^3}{6}s^3 \right) \mathbf{r}(t - T, T) = \mathbf{r}(t - T, T) + \\ &T \dot{\mathbf{r}}(t - T, T) + \frac{T^2}{2} \ddot{\mathbf{r}}(t - T, T) + \frac{T^3}{6} \dddot{\mathbf{r}}(t - T, T) := \mathbf{r}_{caus}(t, T), \end{aligned} \right\} \quad (5)$$

where

$$\mathbf{r}(t - T, T) = \frac{1}{T} \int_{\tau=t-T}^t \mathbf{F}(\tau, \mathbf{x}(\tau)) d\tau, \quad (6)$$

depends on available information $\{\mathbf{x}(\tau)\}_{\tau \in [t-T, t]}$. Given that the new variable $\mathbf{r}_{caus}(t, T)$ (5) can be treated as the "causal approximation" of variable $\mathbf{r}(t, T)$ and the integral representation (4) of the considered dynamics (1) can be locally approximated as

$$\mathbf{x}(t + T) \simeq \mathbf{x}(t) + T \mathbf{r}_{caus}(t, T). \quad (7)$$

Remark 1 Since the right-hand side of (7) contains only information $\{x(\tau)\}_{\tau \in [t-T, t]}$, available up to time t , we can consider the value $x(t + T)$ as the "prediction (or forecasting)" of the process $\{\mathbf{x}(\tau)\}_{\tau \in [0, t]}$ ahead on horizon T .

3 Prediction Algorithm for Models with Incomplete Information: DNN Approach

When the original dynamics $\mathbf{F}(t, \mathbf{x}(t))$ in (3) is completely or partially unknown, we suggest applying the DNN approach [11] which showed nice results being applied to various problems in bio-engineering and the environment science [12, 13].

3.1 DNN Identification Model

Artificial neural networks (ANNs) are thought to be effective modeling tools for non-linear, complicated systems with ambiguous dynamic models. ANNs were first utilized as reliable predictors of various processes with static reliance on input–output data. The time effect must be included in the ANN when it is used to characterize a rough model of time-dependent input–output relationships, which necessitates the reconstruction of a dynamic ANN or the use of *Recurrent Neural Networks* (RNNs) in discrete time or *Differential Neural Networks* (DNNs) in continuous time. DNNs sometimes referred to as *Auto Associative* or *Feedback Networks*, are a subclass of ANNs in which the connections between the input and the output

are organized into a directed cycle. As a result, the network develops an internal state that displays dynamic, temporally dependent behavior. DNN allows the signal to go both forward and backward by including loops in the network design or topology. To achieve the required behavior of this DNN, a particular tuning for the time-dependent weight matrix parameters is realized as a result of such a suggestion. In our scenario, define the single layer DNN model following [11], where the measurable output $x(t)$ is a vector, as

$$\left. \begin{aligned} \frac{d}{dt} \hat{\mathbf{x}}(t) &= A \hat{\mathbf{x}}(t) + \mathbf{b}x_1^{(n)}(t) + \hat{W}(t) \sigma(\hat{\mathbf{x}}(t)) + L[x_1(t) - C^\top \hat{\mathbf{x}}(t)], \\ \hat{\mathbf{x}}(0) &= \mathbf{x}(0) \in \mathbb{R}^n, C^\top = (1, 0, \dots, 0) \in \mathbb{R}^n, \hat{W}(t) \in \mathbb{R}^{n \times p}, \sigma : \mathbb{R}^n \rightarrow \mathbb{R}^p, \end{aligned} \right\} \quad (8)$$

where

- $\sigma^\top(\hat{\mathbf{x}}) = (\sigma_1(\hat{\mathbf{x}}), \sigma_2(\hat{\mathbf{x}}), \dots, \sigma_p(\hat{\mathbf{x}}))$ is the vector with sigmoidal components

$$\sigma_j(\hat{\mathbf{x}}) = \frac{\alpha_j}{1 + \beta_j e^{-\gamma_j^\top \hat{\mathbf{x}}}} + \delta_j, j = 1, \dots, p$$

(α_j, β_j and δ_j are positive scalars and $\gamma_j \in \mathbb{R}^n$ is a weighting vector for the component of $\hat{\mathbf{x}}$);

- $\hat{W}(t)$ is the weight matrix, changing in time according to the *Learning Procedure* (LP)

$$\left. \begin{aligned} \frac{d}{dt} \hat{W}(t) &= K^{-1} P [\mathbf{x}(t) - \hat{\mathbf{x}}(t)] \sigma^\top(\hat{\mathbf{x}}(t)) \\ 0 < K = K^\top \in \mathbb{R}^{n \times n}, 0 < P = P^\top \in \mathbb{R}^{n \times n} \end{aligned} \right\} \quad (9)$$

- The vector $L \in \mathbb{R}^n$ must be selected in such a way that

$$L \in \mathbb{R}^{n \times 1} : A^0(L) = A - LC^\top \text{ is Hurwitz,} \\ \text{spectrum}(A^0(L)) \in \mathbb{C}^-.$$

As it is mentioned in ([11]), a special selection of matrix P we may guarantee a good DNN-approximation (identification) $\hat{\mathbf{x}}(t) \simeq \mathbf{x}(t)$ practically for all $t \geq 0$. The next subsection explains how the algorithms (8) and (9) should be modified to be able to generate a *good prediction trajectory* $\hat{\mathbf{x}}(t + T)$ using only available information $\{\hat{\mathbf{x}}(\tau), \}_{\tau \in [t-T, t]}$.

3.2 DNN Prediction Model

The DNN dynamics (8) in the integral causal format (7) may be represented as

$$\hat{\mathbf{x}}(t + T) = \hat{\mathbf{x}}(t) + T \hat{\mathbf{r}}_{caus}(t, T), \quad (10)$$

where

- the signal $\hat{\mathbf{x}}(t)$ is generated by (8),
- the auxiliary vector $\hat{\mathbf{r}}_{caus}(t, T)$ is defined as

$$\begin{aligned} \hat{\mathbf{r}}_{caus}(t, T) := & \hat{\mathbf{r}}(t - T, T) + T \frac{d}{dt} \hat{\mathbf{r}}(t - T, T) \\ & + \frac{T^2}{2} \frac{d^2}{dt^2} \hat{\mathbf{r}}(t - T, T) + \frac{T^3}{2} \frac{d^3}{dt^3} \hat{\mathbf{r}}(t - T, T) \end{aligned} \tag{11}$$

with

$$\left. \begin{aligned} \hat{\mathbf{r}}(t - T, T) := & \frac{1}{T} \int_{\tau=t-T}^t \hat{\mathbf{F}}(\tau, \mathbf{x}(\tau)) d\tau, \\ \hat{\mathbf{F}}(\tau, \mathbf{x}(\tau)) := & \mathbf{A}\hat{\mathbf{x}}(\tau) + \mathbf{b}x_1^{(n)}(\tau) + \hat{\mathbf{W}}(\tau) \sigma(\hat{\mathbf{x}}(\tau)) + L[x_1(\tau) - \mathbf{C}^\top \hat{\mathbf{x}}(\tau)], \end{aligned} \right\} \tag{12}$$

- the derivatives $x_1^{(m)}(t)$, ($m = 1, \dots, n$) and $\frac{d^k}{dt^k} \hat{\mathbf{r}}(t - T, T)$, ($k = 1, 2, 3$) are calculated recurrently based on "super-twist algorithm" ([9], [16]). To differentiate time function $f(t)$, the super-twisting controller is designed to reduce the error $s(t)$ ($s = x - f$) between its input $f(t)$ and output $x(t)$ to zero:

$$\left. \begin{aligned} \dot{x}(t) = & -\alpha \sqrt{|s(t)|} \text{sign}(s(t)) + y(t), \\ \dot{y}(t) = & -M \text{sign}(s(t)) \\ |\ddot{f}| < & F_0, M > F_0 \end{aligned} \right\} \tag{13}$$

The error $s(t)$ is reduced to zero after a finite time interval t_0 and state component $y(t)$ is equal to the first-time derivative of a function $f(t)$, namely, $y(t) = \frac{d}{dt} f(t)$ for all $t \geq t_0$. If $f(t)$ is corrupted by bounded noise $|s(t)| \leq \Delta = \text{const}$, then an upper bound of the differentiation, the error is estimated by inequality

$$\left| y(t) - \frac{d}{dt} f(t) \right| \leq \alpha_1 \Delta + \alpha_2 \sqrt{\Delta}, \alpha_1, \alpha_2 - \text{positive constants.}$$

3.3 DNN Predictor with Slow and Fast Components

There are several systems whose trajectories can be understood as the overlapping of signals formed with the combination of slow and fast components. Such systems are also known as multi-rate system that appears naturally in mobile robotics [4], chemical [17] and biochemical [6] reactions, evolution of medical sicknesses [14], evolution of ecosystems animal populations [5] and many others. The same type of combined dynamics is valid for describing the evolution of both infected and deceased persons suffering of the Covid-19 sickness.

The developed DNN structure with mixed (slow and fast) learning scheme could be useful to represent the dynamics of COVID-19. Such a fact can be justified considering that the evolution of infected and deceased persons can be represented as the combination of a slow dynamics defined by the seasonal variations and a fast evolution which corresponds to the daily evolution. One may notice that such multi-rate dynamics has not been considered before in the design of non-parametric identifiers based on differential neural networks, which is indeed a contribution of this study.

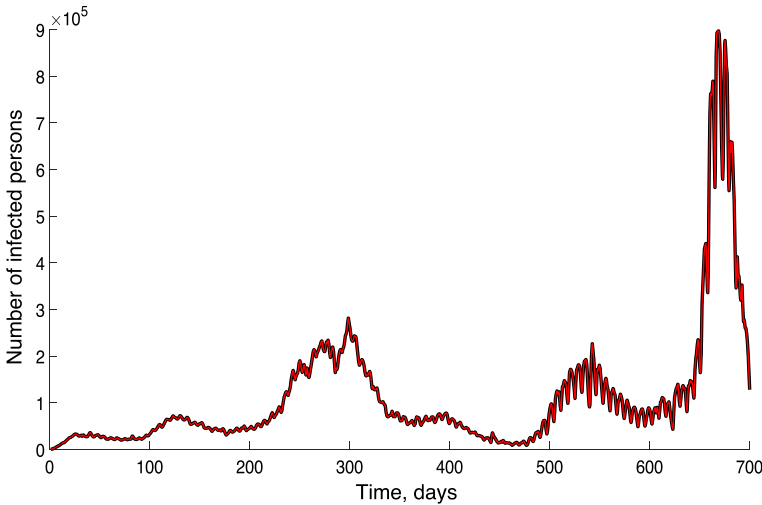


Fig. 1 Comparative evolution of the slow part of the infected cases over time using the corresponding DNN with its state $\mathbf{x}_{slow}(t)$

3.3.1 Slow Predictive Component

Based on available data $\{x(\tau)\}_{\tau \in [0,t]}$ let us reconstruct a "slow" trajectory $\{x_{slow}(\tau)\}_{\tau \in [0,t]}$ defined as the *best least squares polynomial approximation* of a given order N , that is,

$$\left. \begin{aligned}
 x_{slow}(t) &= \sum_{i=0}^N \bar{c}_i t^i, \\
 \bar{\mathbf{c}} &= \arg \min_{\mathbf{c} \in \mathbb{R}^{N+1}} \int_{\tau=0}^t \left(x(\tau) - \sum_{i=0}^N c_i \tau^i \right)^2 d\tau \\
 &= \left(\int_{\tau=0}^t x(\tau) \boldsymbol{\theta}(\tau) \boldsymbol{\theta}^\top(\tau) d\tau \right)^{-1} \int_{\tau=0}^t x(\tau) \boldsymbol{\theta}(\tau) d\tau, \\
 \bar{\mathbf{c}}^\top &:= (\bar{c}_0, \dots, \bar{c}_N) \in \mathbb{R}^{N+1}, \boldsymbol{\theta}^\top(\tau) := (1, \tau, \tau^2, \dots, \tau^N) \in \mathbb{R}^{N+1}
 \end{aligned} \right\} \tag{14}$$

The behavior of trajectory $\mathbf{x}_{slow}(t)$ is shown in Fig. 1 for COVID-19 case.

Then, as in (10), (11) and (12), define $\hat{\mathbf{x}}_{slow}(t+T)$ as

$$\hat{\mathbf{x}}_{slow}(t+T) = \hat{\mathbf{x}}_{slow}(t) + T \hat{\mathbf{r}}_{caus}^{slow}(t, T), \tag{15}$$

where

- $\hat{\mathbf{x}}_{slow}(t)$ is generated by the following DNN model:

$$\left. \begin{aligned}
 \frac{d}{dt} \hat{\mathbf{x}}_{slow}(t) &= A \hat{\mathbf{x}}_{slow}(t) + \mathbf{b} x_{1,slow}^{(n)}(t) + \\
 &\quad \hat{W}_{slow}(t) \sigma(\hat{\mathbf{x}}_{slow}(t)) + L [x_{1,slow}(t) - C^\top \hat{\mathbf{x}}_{slow}(t)], \\
 \frac{d}{dt} \hat{W}_{slow}(t) &= K^{-1} P (\mathbf{x}_{slow}(t) - \hat{\mathbf{x}}_{slow}(t)) \sigma^\top(\hat{\mathbf{x}}_{slow}(t)) \\
 \hat{\mathbf{x}}_{slow}(0) &= \mathbf{x}(0) \in \mathbb{R}^n, C^\top = (1, 0, \dots, 0) \in \mathbb{R}^n, \\
 \hat{W}_{slow}(t) &\in \mathbb{R}^{n \times p}, \sigma : \mathbb{R}^n \rightarrow \mathbb{R}^p,
 \end{aligned} \right\} \tag{16}$$

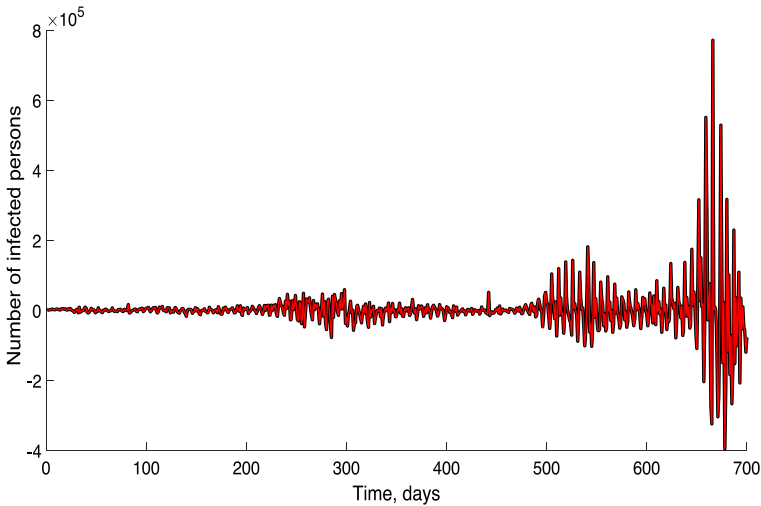


Fig. 2 Comparative evolution of the faster part of the infected cases over time using the corresponding DNN with its state $\mathbf{x}_{fast}(t)$

- the auxiliary vector $\hat{\mathbf{r}}_{caus}^{slow}(t, T)$ is defined as

$$\left. \begin{aligned} \hat{\mathbf{r}}_{caus}^{slow}(t, T) := & \hat{\mathbf{r}}^{slow}(t - T, T) + T \frac{d}{dt} \hat{\mathbf{r}}^{slow}(t - T, T) + \\ & \frac{T^2}{2} \frac{d^2}{dt^2} \hat{\mathbf{r}}^{slow}(t - T, T) + \frac{T^3}{2} \frac{d^3}{dt^3} \hat{\mathbf{r}}^{slow}(t - T, T), \end{aligned} \right\} \quad (17)$$

with

$$\left. \begin{aligned} \hat{\mathbf{r}}^{slow}(t - T, T) := & \frac{1}{T} \int_{\tau=t-T}^t \hat{\mathbf{F}}^{slow}(\tau, \mathbf{x}_{slow}(\tau)) d\tau, \\ \hat{\mathbf{F}}^{slow}(\tau, \mathbf{x}_{slow}(\tau)) := & A \hat{\mathbf{x}}_{slow}(\tau) + \mathbf{b} x_{1,slow}^{(n)}(\tau) + \\ & \hat{W}(\tau) \sigma(\hat{\mathbf{x}}_{slow}(\tau)) + L[x_{1,slow}(\tau) - C^T \hat{\mathbf{x}}_{slow}(\tau)], \end{aligned} \right\} \quad (18)$$

3.3.2 Fast Predictive Component

Define $x_{fast}(t)$ as

$$x_{fast}(t) := x(t) - x_{slow}(t) \quad (19)$$

The behavior of trajectory $\mathbf{x}_{fast}(t)$ is shown in Fig. 2 for COVID-19 case. Then, as in (16), (17) and (18), define $\hat{\mathbf{x}}_{fast}(t + T)$ as

$$\hat{\mathbf{x}}_{fast}(t + T) = \hat{\mathbf{x}}_{fast}(t) + T \hat{\mathbf{r}}_{caus}^{fast}(t, T), \quad (20)$$

where

- $\hat{\mathbf{x}}_{fast}(t)$ is generated by the following DNN model:

$$\left. \begin{aligned}
 \frac{d}{dt} \hat{\mathbf{x}}_{fast}(t) &= A \hat{\mathbf{x}}_{fast}(t) + \mathbf{b}x_{1,fast}^{(n)}(t) + \\
 \hat{W}_{fast}(t) \sigma(\hat{\mathbf{x}}_{fast}(t)) + L[x_{1,fast}(t) - C^\top \hat{\mathbf{x}}_{fast}(t)], \\
 \frac{d}{dt} \hat{W}_{fast}(t) &= K^{-1}P(\mathbf{x}_{fast}(t) - \hat{\mathbf{x}}_{fast}(t))\sigma^\top(\hat{\mathbf{x}}_{fast}(t)) \\
 \hat{\mathbf{x}}_{fast}(0) &= \mathbf{x}(0) \in \mathbb{R}^n, C^\top = (1, 0, \dots, 0) \in \mathbb{R}^n, \\
 \hat{W}_{fast}(t) &\in \mathbb{R}^{n \times p}, \sigma : \mathbb{R}^n \rightarrow \mathbb{R}^p,
 \end{aligned} \right\} \tag{21}$$

- the auxiliary vector $\hat{\mathbf{r}}_{caus}^{fast}(t, T)$ is defined as

$$\left. \begin{aligned}
 \hat{\mathbf{r}}_{caus}^{fast}(t, T) &:= \hat{\mathbf{r}}^{fast}(t - T, T) + T \frac{d}{dt} \hat{\mathbf{r}}^{fast}(t - T, T) + \\
 \frac{T^2}{2} \frac{d^2}{dt^2} \hat{\mathbf{r}}^{fast}(t - T, T) + \frac{T^3}{2} \frac{d^3}{dt^3} \hat{\mathbf{r}}^{fast}(t - T, T),
 \end{aligned} \right\} \tag{22}$$

with

$$\left. \begin{aligned}
 \hat{\mathbf{r}}^{fast}(t - T, T) &:= \frac{1}{T} \int_{\tau=t-T}^t \hat{\mathbf{F}}^{fast}(\tau, \mathbf{x}_{fast}(\tau)) d\tau, \\
 \hat{\mathbf{F}}^{fast}(\tau, \mathbf{x}_{fast}(\tau)) &:= A \hat{\mathbf{x}}_{fast}(\tau) + \mathbf{b}x_{1,fast}^{(n)}(\tau) + \\
 \hat{W}(\tau) \sigma(\hat{\mathbf{x}}_{fast}(\tau)) + L[x_{1,fast}(\tau) - C^\top \hat{\mathbf{x}}_{fast}(\tau)].
 \end{aligned} \right\} \tag{23}$$

3.4 Joint Slow and Fast Predictor

In this paper, we use more *advance predictor*, consisting in two components: slow $\hat{\mathbf{x}}_{slow}(t + T)$ and fast $\hat{\mathbf{x}}_{fast}(t + T)$, namely,

$$\hat{\mathbf{x}}(t + T) = \hat{\mathbf{x}}_{slow}(t + T) + \hat{\mathbf{x}}_{fast}(t + T). \tag{24}$$

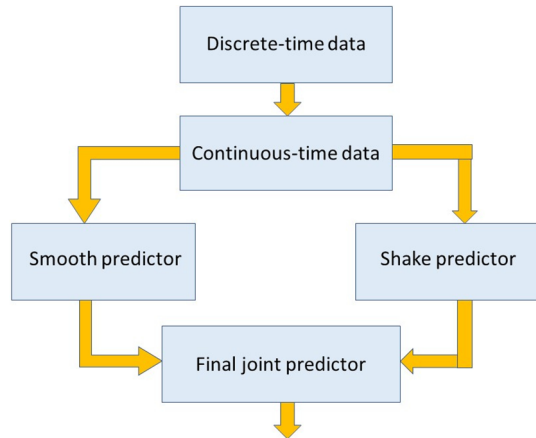
4 Structure of Numerical Procedure

The suggested predictive numerical structure consists of the following steps:

1. Based on given discrete-time data $\{x(k)\}_{k \in [0, 1, \dots, \mathcal{K}]}$, where $x(k)$ corresponds to the data value at day k , and applying a spline approximation (herein examples, we use the spline of 15-th order) we construct the continuous-time curve $\{x(\tau)\}_{\tau \in [0, t]}$ where $t = \mathcal{K}\Delta$ (Δ is time interval between discrete data).
2. Then using (14) and (19), based on the obtained curve $\{x(\tau)\}_{\tau \in [0, t]}$ we need to construct the slow $x_{slow}(t)$ and fast $x_{fast}(t)$ trajectories.
3. Applying the procedures (16), (17) and (18) we obtain the slow predictive curve $\hat{\mathbf{x}}_{slow}(t + T)$ (15).
4. Then applying the procedures (20), (22) and (23) we obtain the fast predictive curve $\hat{\mathbf{x}}_{fast}(t + T)$ (15).
5. The last step is to construct the final predictive curve $\hat{\mathbf{x}}(t + T)$ (24) for desired T (for example, taking $T = 60, 90, 120$ days on the COVID-19 prediction).

The corresponding block scheme is shown in Fig. 3.

Fig. 3 Flow diagram describing how the proposed forecasting evolution is derived using the tools included in the toolbox



The developed algorithm was implemented accordingly to the following pseudocode.

1. Load information corresponding to infected or deceased patients suffered from COVID-19 sickness
2. Interpolate the loaded information using a third order spline strategy
3. Implement a p -th order low-pass filter with a finite-impulse response strategy using a cut-off frequency of 0.5 Hz. This frequency was determined using the collected information. The value of p -th order is fixed to 7 considering the evolution of Covid information.
4. Filter the loaded information separating the slow and fast components of the infected or deceased datasets, according to the selected cut-off frequency.
5. Develop the slow learning algorithm in the first differential neural network implemented as a non-parametric identifier.
6. Develop the fast learning algorithm in the first differential neural network implemented as a non-parametric identifier.
7. Divide the information considering the training period and the complementary validation period.
8. Evaluate both the slow and fast identifiers to reproduce the information corresponding to the information considered in the training period.
9. Repeat the identification task until the least mean square error of the identification error for both the slow and the fast learning is smaller than a given threshold value ε .
10. Once the expected quality of training is expected, recover the values of the weights produced during this part of the process for both the slow and fast evaluations.
11. Develop the numerical simulation of the differential neural network working as the predictor using two models using the recovered weights from the slow and the fast evolution of the training algorithms.
12. Add the results of the slow and fast predictors to reconstruct the information during the prediction period.
13. Compare if possible, the information obtained from the Covid statistics during the prediction period with respect to the obtained data during the evaluation of the added identifier.
14. Determine the Least Mean Square Error and the Maximum Error for the predicted period, if possible to characterize the quality of the prediction task.

5 Seventy Days Prediction of Infections and Deaths for Different Countries

This research uses a publicly available dataset “2019 Novel Coronavirus Data Repository-published by Johns Hopkins University Center for Systems Science and Engineering (JHU CSSE) available at: <https://github.com/CSSEGISandData/COVID-19>. Models achieved and the code used in their generation are available in a repository located at: <https://github.com/RitehAIandRobot/COVID-19-MLP>. This information is proposed in COVID-19 MLP, Riteh AI and Robotics Group, 2020, <https://github.com/RitehAIandRobot/COVID-19-MLP>.

The presented set of numerical simulations considered a temporal horizon of 70 days. All the selected parameters were obtained using the Hurwitz conditions for $A - LC^T$. The values for the parameters considered in the activation functions were obtained with a uniform distribution for the exponential term, unitary gain with a fixed offset to 0.5. Hence, the parameters used for solving the numerical simulation for this the study was the following:

$$A = 1.0 \cdot 10^{-2} \cdot \begin{bmatrix} -25 & 0 & 0 & 0 \\ -8 & -32 & 0 & 0 \\ -12 & -11 & -44 & 0 \\ -13 & -13 & -12 & -52 \end{bmatrix} \tag{25}$$

The number of sigmoidal functions (artificial neurons in the DNN) used for the identification process was 9600. The parameters in the sigmoidal functions were $\alpha_j = 1$ for all $j = 1, \dots, 9600$. The parameter in the denominator are $\beta_j = 0.05$ and $\beta_j = 0.08$ for all $j = 1, \dots, 9600$. The period T was fixed to 10 days. All the initial conditions were fixed as random values between 0 and 1. These selections were obtained using a trial and testing method that effectively estimated the number of infected and deceased persons with Sars-Cov2 virus. These estimations were evaluated using the collected information reported by the World Health Organization.

The values of matrices K , P , and L are as follows:

$$\left. \begin{aligned} K &= \begin{bmatrix} 25.0 & 1.2 & -1.5 & 2.3 \\ 1.2 & 32.0 & -1.1 & -1.3 \\ -1.5 & -1.1 & 44 & 1.2 \\ 2.3 & -1.3 & 1.2 & 52 \end{bmatrix}, \\ P &= \begin{bmatrix} 2.5 & 0.8 & -1.2 & -0.3 \\ 0.8 & 5.2 & -1.1 & -0.1 \\ -1.2 & -1.1 & 6.4 & -1.2 \\ -0.3 & -0.1 & -1.2 & 9.2 \end{bmatrix}, \\ L &= [-15 \ -28 \ -32 \ -53]^T. \end{aligned} \right\} \tag{26}$$

5.1 Turkey

Figure 4 shows the comparison of estimated data evolution for infected people in Turkey. The comparison of trajectories confirms at first glance the effectiveness of the proposed DNN-based forecasting considering a period of estimation of 70 days. Moreover, it shows the effective estimation of the forecast information.

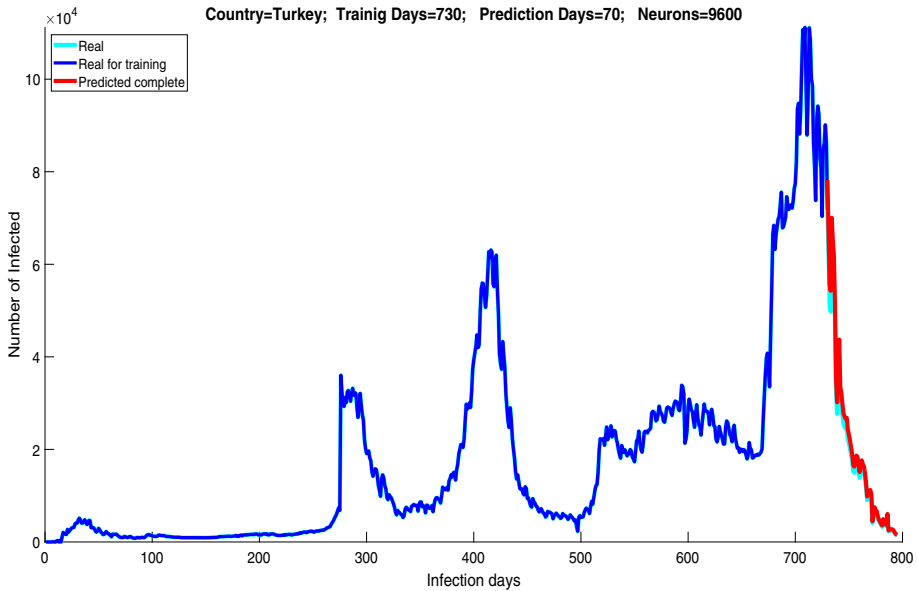


Fig. 4 Prediction results with future estimation for the infected people detected in Turkey using the proposed DNN-based predictor

Figure 5 depicts the evolution of the predicted data for deceased people in Turkey using the proposed multi-rate identifier. In this case, there is a comparison considering the estimated data and the one corresponding to the actual data.

5.2 USA

Figure 6 shows the comparison of estimated data evolution for infected people in the United States of America. The comparison of trajectories confirms at first glance the effectiveness of the proposed DNN-based forecasting considering a period of 70 days.

Figure 5 depicts the evolution of the predicted data for deceased people in the United States of America using the proposed multi-rate identifier. In this case, there is a comparison considering the estimated data and the one corresponding to the actual data.

All the previous results confirm that the proposed forecaster is based on the dual configuration of DNN. Moreover, the proposed technique can be easily implemented in different forecast problems taking advantage of the generalized formulation presented here.

For both studied cases, we included here some methods used for comparison including a traditional recurrent neural network (RNN), a Long-Short term memory (LSTM), and a gated network unit (GNU). These networks were considered for comparison taking into account the significant outcomes shown before as potential predictors of complex time dependent information. We have presented two tables (one per infected and one per deceased persons) comparing some quality measurements, including the least mean square evaluation for the signals corresponding to the evolution of infected and deceased persons during the COVID outbreak (Tables 1 and 2). With the aim of introducing a fair comparison, the number of flops used for each of the prediction tasks was also estimated. These results confirm the

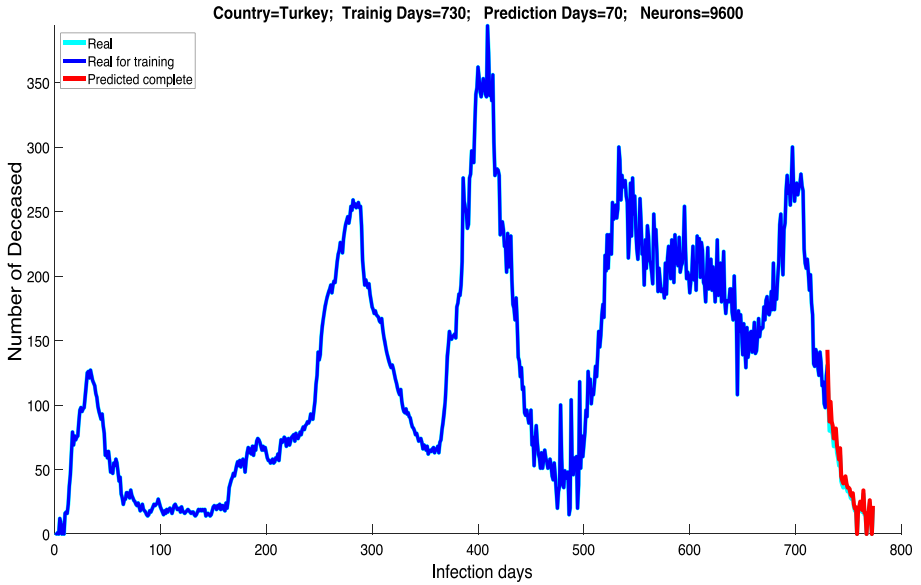


Fig. 5 Prediction results with future estimation for the deceased people detected in Turkey using the proposed DNN-based predictor

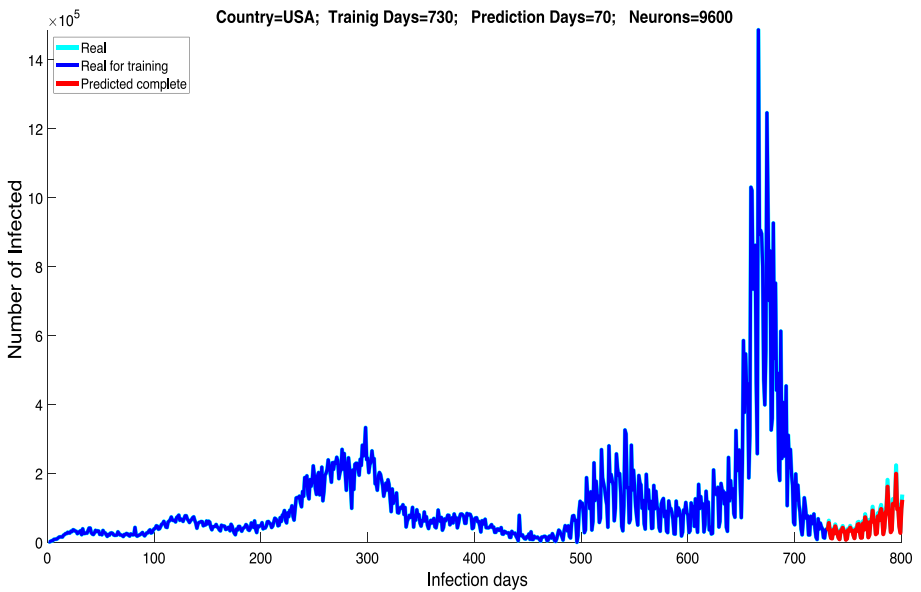


Fig. 6 Prediction results with future estimation for the infected people detected in the United States of America using the proposed DNN-based predictor

Table 1 Comparison results for infected persons showing the prediction outcomes for the infected persons and the flops used to perform the task

Method	LMSE	MEPP	Flops
RNN	193.5	398.0	7.8×10^7
LSTM	137.5	245.0	1.8×10^7
GNU	167.4	267.0	3.8×10^7
Slow-fast method	54.5	78.0	3.2×10^8

Table 2 Comparison results for deceased persons showing the prediction outcomes for the infected persons and the flops used to perform the task

Method	LMSE	MEPP	Flops
RNN	45.5	48.0	5.4×10^7
LSTM	65.3	67.0	4.3×10^7
GNU	76.4	78.0	1.9×10^7
Slow-fast method	34.3	24.0	3.0×10^8

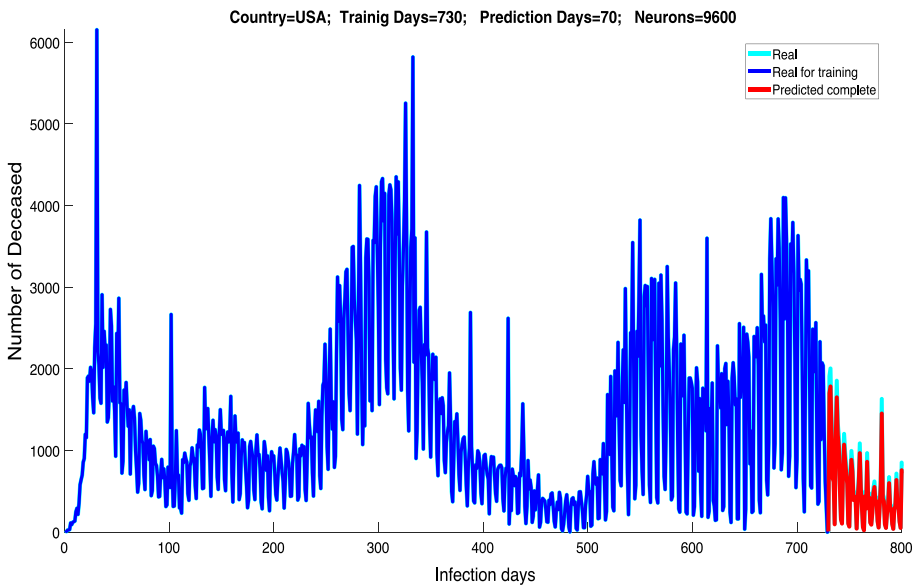


Fig. 7 Prediction results with future estimation for the deceased people detected in the United States of America using the proposed DNN-based predictor

advances generated by applying the proposed predictor based on the combined learning method introduced in this study.

The proposed outcomes shown in the previous tables confirm the benefits of the proposed methodology, including the prediction quality, as well as the convergence conditions (noticing the maximum error value). However, the augmented number of flops required by the methodology considered in this study still requires some work to improve the prediction abilities. Moreover, showing the better least mean square errors obtained with the proposed methodology highlights the benefit of introducing the mixed learning with slow and fast dynamics.

6 Conclusions

- In this paper, it is shown that time-series data may be effectively modeled by a Differential Neural Network (DNN) with time-varying weights matrix parameters whose dynamics are governed by special *Learning laws* containing slow and fast components;
- This study also demonstrates one of the possible applications of the suggested technique to COVID-19 epidemic prediction, where we suppose that the current data set represents the output of some dynamic system, governed by a nonlinear ordinary differential equation; this method has been evaluated for two nations' databases (Turkey and the USA) and has demonstrated great performances (70 days of forecast).

Acknowledgements The paper was prepared under the financial support of the Automatic Control Department at CINVESTAV-IPN, Mexico.

Data availability The datasets generated during and/or analyzed during the current study are available from the corresponding author upon reasonable request.

Declarations

Conflict of interest All the authors declare no conflict of interest.

References

1. Adam D (2020) Special report: the simulations driving the world's response to COVID-19. *Nature* 580(7803):316–318
2. Anirudh A (2020) Mathematical modeling and the transmission dynamics in predicting the Covid-19 - What next in combating the pandemic. *Infect Disease Model* 5:366–374
3. Balcan D et al (2010) Modeling the spatial spread of infectious diseases: the global epidemic and mobility computational model. *J Comput Sci* 1(3):132–145
4. da Costa Barros R, Nascimento TP (2021) Robotic mobile fulfillment systems: a survey on recent developments and research opportunities. *Robot Auton Syst* 137:103729
5. Deangelis DL, Mooij WM (2005) Individual-based modeling of ecological and evolutionary processes. *Ann Rev Ecol Evol Syst* 147–168
6. Goutsias John (2007) Classical versus stochastic kinetics modeling of biochemical reaction systems. *Biophys J* 92(7):2350–2365
7. Haykin S (2008) *Neural networks and learning machines*, 3rd edn. Prentice Hall, Upper Saddle River
8. Jewell NP, Lewnard JA, Jewell BL (2020) Predictive mathematical models of the COVID-19 pandemic: underlying principles and value of projections. *JAMA* 323(19):1893–1894. <https://doi.org/10.1001/jama.2020.6585>
9. Levant A (2003) Higher-order sliding modes, differentiation and output-feedback control. *Int J Control* 76(9):924–941
10. Priyanka R, Kumari A, Sood M (2021) Implementation of simple RNN and LSTMs based prediction model for coronavirus disease (Covid-19). In: IOP Conf. Ser.: Mater. Sci. Eng., 1022 012015
11. Poznyak A, Sanchez E, Yu W (2001) *Differential neural networks for robust nonlinear control: identification, state estimation, and trajectory tracking*. World Scientific, Singapore
12. Poznyak T, Chairez I, Poznyak A (2019) *Ozonation and biodegradation in environmental engineering: dynamic neural network approach*. Elsevier, Amsterdam-NY
13. Poznyak A, Chairez I, Poznyak T (2019) A survey on artificial neural networks application for identification and control in environmental engineering: biological and chemical systems with uncertain models. *Annu Rev Control* 48:250–272
14. Reason JT (1978) Motion sickness adaptation: a neural mismatch model. *J R Soc Med* 71(11):819–829
15. Santosh KC (2020) COVID-19 prediction models and unexploited data. *J Med Syst* 44(9):1–4
16. Utkin V, Poznyak A, Orlov YV, Polyakov A (2020) Road map for sliding mode control design, Springer-Briefs in mathematics. Springer, Berlin. <https://doi.org/10.1007/978-3-030-41709-3>

17. Vora Nishith, Daoutidis Prodromos (2001) Nonlinear model reduction of chemical reaction systems. *AICHE J* 47(10):2320–2332
18. Wynants L, Van Calster B, Collins GS, Riley RD, Heinze G, Schuit E, van Smeden M (2020) Prediction models for diagnosis and prognosis of covid-19: a systematic review and critical appraisal. *BMJ* 369
19. Zeb A, Alzahrani E, Suat Erturk V, Zaman G (2020) Mathematical model for coronavirus disease 2019 (COVID-19) containing isolation class. *Hindawi BioMed Rese Int*. <https://doi.org/10.1155/2020/3452402>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.