


ORIGINAL RESEARCH

Gravitational search algorithm-extreme learning machine for COVID-19 active cases forecasting

Boyu Huang¹  | Youyi Song² | Zhihan Cui¹ | Haowen Dou¹ | Dazhi Jiang¹ | Teng Zhou^{2,3}  | Jing Qin²

¹Department of Computer Science, Shantou University, Shantou, China

²Centre for Smart Health, School of Nursing, The Hong Kong Polytechnic University, Hong Kong, China

³School of Cyberspace Security, Hainan University, Haikou, China

Correspondence

Teng Zhou and Jing Qin.
Email: teng.zhou@polyu.edu.hk and harry.qin@polyu.edu.hk

Funding information

Natural Science Foundation of Guangdong Province, Grant/Award Number: 2022A1515011590; Project of Strategic Importance of The Hong Kong Polytechnic University, Grant/Award Number: 1-ZE2Q; National Natural Science Foundation of China, Grant/Award Number: 61902232; Li Ka Shing Foundation, Grant/Award Number: 2020LKSFG05D

Abstract

Corona Virus disease 2019 (COVID-19) has shattered people's daily lives and is spreading rapidly across the globe. Existing non-pharmaceutical intervention solutions often require timely and precise selection of small areas of people for containment or even isolation. Although such containment has been successful in stopping or mitigating the spread of COVID-19 in some countries, it has been criticized as inefficient or ineffective, because of the time-delayed and sophisticated nature of the statistics on determining cases. To address these concerns, we propose a GSA-ELM model based on a gravitational search algorithm to forecast the global number of active cases of COVID-19. The model employs the gravitational search algorithm, which utilises the gravitational law between two particles to guide the motion of each particle to optimise the search for the global optimal solution, and utilises an extreme learning machine to address the effects of nonlinearity in the number of active cases. Extensive experiments are conducted on the statistical COVID-19 dataset from Johns Hopkins University, the MAPE of the authors' model is 7.79%, which corroborates the superiority of the model to state-of-the-art methods.

KEYWORDS

adaptive systems, behavioural sciences computing, big data, data analysis

1 | INTRODUCTION

An outbreak of *unexplained pneumonia* in Wuhan, Hubei Province, China, in early December 2019, has been confirmed as an acute respiratory infection caused by a novel 2019 coronavirus infection [1–3]. Although the source of COVID-19 transmission has not been determined, Zhou et al. [4] pointed out the possibility that the virus was transmitted from bats to humans. Based on the assessment, the World Health Organisation (WHO) believes that the current outbreak of new coronavirus pneumonia can be described as a global pandemic [5]. With the global spread of COVID-19, many countries have declared a state of emergency prevention and control, and have introduced cross-border travel restrictions one after another, which has caused a large number of deaths and significant economic devastation. As countries assess the severity of the

outbreak in their regions mainly based on the number of active cases, accurate prediction of active COVID-19 cases is an essential technique for deciding on outbreak prevention and control measures to reduce the number of COVID-19 infections. The quantity of active cases is not only influenced by the randomness of infection between virus carriers and normal individuals, but also displays seasonal variations [6]. Owing to the variability and stochasticity of infection between individuals [7, 8], it is still imperative to construct a model that can accurately forecast.

The virus is transmitted from person to person primarily through respiratory droplets [7–9] and causes a range of symptoms and severe sequelae [10–12]. Nevertheless, the exact virological and epidemiological characteristics of this third zoonotic coronavirus, including transmissibility and mortality, are not known. Deep learning has the ability to learn and

This is an open access article under the terms of the [Creative Commons Attribution-NonCommercial-NoDerivs](https://creativecommons.org/licenses/by-nc-nd/4.0/) License, which permits use and distribution in any medium, provided the original work is properly cited, the use is non-commercial and no modifications or adaptations are made.

© 2023 The Authors. *IET Software* published by John Wiley & Sons Ltd.

model nonlinear complex relationships which has received interest and attention in various fields [13–16]. Nevertheless, with such a significant number of active cases worldwide, training deep learning models would be time-consuming and vulnerable to overfitting [17]. Therefore, we need to formulate a model that can solve these problems properly to forecast COVID-19 active cases.

Extreme learning machine (ELM) is a feedforward neural network first proposed by Huang et al. [18] in 2006. It is illustrated that ELM has excellent generalisation performance as well as extremely fast learning ability without gradient-based backpropagation to adjust the weights, and instead sets the weights by the Moore-Penrose (MP) generalised inverse, so it can well overcome the problem of difficult training with large amount of data [19, 20]. It is further proved that if the activation function of the hidden layer is infinitely differentiable on any interval, the input weight and the hidden layer threshold can be set randomly before training and remain unchanged during the training. ELM is currently applied not only to regression and fitting problems [21, 22], but also to classification [23], pattern recognition [24] and other fields. At the same time, a variety of improved methods and strategies have been mentioned [25, 26], so that the performance of ELM has also been greatly improved, and its importance is increasingly reflected.

Nevertheless, ELM generates the weight matrix of the input layer and the hidden layer bias values by initializing them in a random way. These parameter setting situations play a significant role in the final prediction performance of the model. In addition, the nonlinear active cases data leads to the possibility that the model may suffer from performance degradation in the face of samples that do not appear during the training process due to the lack of generalisation ability. To address this limitation, the GSA-ELM algorithm proposed in this paper can solve the above problems. We utilise the Gravitational Search Algorithm (GSA) [27] to search for the most optimal weights and biases of the extreme learning machine in cases forecasting, referred to as GSA-ELM. This model not only reduces the complexity of the network to avoid overfitting, but also ensures the reliability and stability of the prediction results and improves the competitiveness relative to other methods. We summarise the principal contributions of this work as follows.

- We optimise the weights and biases by gravitational search algorithm to improve the performance of the extreme learning machine.
- We innovatively applied deep learning methods to the prediction of COVID-19 and were capable of successfully coping with the complexities of nonlinearity.
- We evaluate our hybrid learning model on a benchmark set and compare it with several state-of-the-art machine learning forecasting models to demonstrate the superiority of our model.

The rest of the paper is structured as follows. The second part is the methodology, and the third part is an empirical study of real-world data from Johns Hopkins University statistics in the United States. Then comes the conclusion.

2 | RELATED WORKS

Simulation of epidemics. Several epidemiological and clinical characterisation studies have been conducted on patients with this virus to analyse its biological features and viral pathogenesis [28–31], this will help medical practitioners to develop vaccines faster and effectively prevent the spread of this virus in the population. Anastassopoulou et al. [32] made a preliminary prediction of the evolution of the outbreak by means of data modeling. Susceptible Infected Susceptible (SIS) [33, 34], Susceptible Infected Recovered (SIR) [35] and Susceptible Exposed Infected Recovered (SEIR) [36, 37] models provide an alternative approach to epidemic simulation and many research works have been reported. The results show that those SIS, SIR and SEIR models can reflect the dynamics of different epidemics. Meanwhile, these models have been used in COVID-19 [38, 39].

Optimisation algorithms. There are a large number of many excellent optimisation schemes being applied to solve practical problems. Binary versions for RSO have not been created for binary optimisation problems. Awadallah et al. [40] proposed an enhanced binary version of the Rat Swarm Optimiser (RSO) [41] to handle the Feature Selection (FS) problem, and the amazing achievement proved the feasibility of the proposed RSO version. Thawkar et al. [42] proposed a hybrid feature selection method based on the Butterfly Optimisation Algorithm (BOA) [43] and Ant Lion Optimiser (ALO) [44] for breast cancer prediction, which effectively improved the optimisation and classification accuracy. To change the low exploration capability of the traditional Whale Optimisation Algorithm (WOA) [45], Chakraborty et al. [46] studied to provide mWOAPR. A novel variant version improves the exploration capability of the algorithm while balancing the global and local search functions, and successfully applied it to solve image segmentation problems. However, none of these studies have considered the predictive potential of the hybrid learning model in epidemiology, and this study is the first to apply the model to the task of predicting active cases of COVID-19.

Heuristic algorithms. Heuristic algorithms are proposed relative to optimisation algorithms, where the optimal algorithm for a problem seeks the optimal solution for each instance of that problem. At the present stage, heuristic algorithms are dominated by natural body-like algorithms, which have achieved great success. Typical works include Li et al. [47] which proposed the Slime Mold Algorithm (SMA) as an algorithm inspired by the biomotor behavior of simulated slime molds. Through studying the behavioral pattern of slime mold single cell growth and analysing the characteristics of its simulated behavior applying it to computer simulations can lead to optimised results. Tu et al. [48] proposed the Colony Predation Algorithm (CPA) following the strategy used by animal hunting groups, using success rate to adjust the strategy and simulate the selective abandonment behavior of hunting animals. It shows competitive, superior performance in different search environments. Then, the Harris Hawk Optimisation (HHO) algorithm designed by Heidari et al. [49]

achieves population evolution through mathematical modeling of different predation strategies of Harris hawks, with a strong algorithm for finding superiority and without tedious tuning of parameters.

3 | METHODOLOGY

In this section, we first construct the extreme learning machine (ELM) to predict the quantity of active cases. Then, we utilise the gravitational search algorithm to globally optimise the combination of parameters for the ELM.

3.1 | Extreme learning machine

Extreme Learning Machine (ELM) is a novel fast learning algorithm in the neural network structure, which is a forward propagating neural network. For traditional neural networks, especially single hidden layer feedforward neural networks (SLFNs), ELM can reduce the amount of model operations by randomly initialising the input weights and biases [50] and no longer needing to adjust them after they are set. In addition, the connection weights between the implicit and output layers do not need to be adjusted iteratively, but instead are determined by solving a system of equations. The experiments in [51] prove that the algorithm not only has high generalisation ability to guard against overfitting, but also can outperform traditional machine learning algorithms while guaranteeing learning accuracy. A three-layer structure of ELM is demonstrated in Figure 1.

An overview of the primary philosophy of ELM is provided below. For ELM with a three-layer structure, suppose there are Z training samples $(x_i, t_i)_{i=1}^Z$. Here, $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T \in \mathbb{R}^n$ and $t_i = [t_{i1}, t_{i2}, \dots, t_{im}]^T \in \mathbb{R}^m$ represent the input data and the ground truth of the i th sample, respectively. This neural network with a single hidden layer can be represented as:

$$\sum_{j=1}^k \beta_j g(\omega_j^T \cdot x_i + b_j) = t_i, i = 1, \dots, Z, \quad (1)$$

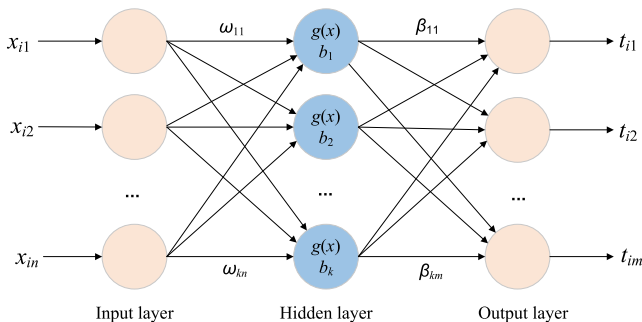


FIGURE 1 Extreme learning machine. The forward propagating neural network has only one hidden layer, whose parameters include input weights ω , output weights β , and hidden layer biases (b).

where $\omega_j = [\omega_{j1}, \omega_{j2}, \dots, \omega_{jk}]^T$ is the input weight vector that links the input nodes to the j th hidden neuron and k is the number of neurons in the hidden layer. $g(x)$ is the activation function of the hidden layer and b_j denotes the bias of the j th hidden neuron. We suppose $h_{ij} = g(\omega_j^T \cdot x_i + b_j)$, where $\omega_j^T \cdot x_i$ is the inner product of ω_j^T and x_i , representing the input value of the j th hidden neuron. Further express Equation (1) in matrix format as:

$$H\beta = T, \quad (2)$$

where $H = \{h_{ij}\}_{i=1, \dots, Z, j=1, \dots, k}$ indicates the output matrix of the implied layer. The matrix $\beta = [\beta_1, \beta_2, \dots, \beta_k]^T$ represents the output weights of this network, in which $\beta_j = [\beta_{j1}, \beta_{j2}, \dots, \beta_{jm}]^T \in \mathbb{R}^m, j = 1, \dots, k$ is the weight vector connecting the j th hidden neuron to the output nodes. $T = [t_1, t_2, \dots, t_Z]^T$ represents the expected output of the network.

In the ELM algorithm, once the input weights and the bias values of the hidden layers are determined randomly, the matrix H is also identified, and training a single hidden layer neural network can be converted into a problem of solving a linear system. The output weight matrix can be obtained by the following equation:

$$\hat{\beta} = H^\dagger T, \quad (3)$$

where H^\dagger is derived by solving the Moore-Penrose (MP) generalised inverse by H .

3.2 | Hybrid GSA-ELM algorithm

ELM inevitably has drawbacks in the learning process. The random selection of its parameters leads to the generation of a series of non-optimal parameters, and these parameter setting situations play an important role in the final prediction performance of the model. This makes the number of required implicit layer nodes more than the traditional learning algorithm, which affects its generalisation performance and leads to the pathological state of the system. Only the information of the input parameters is used in the learning process for computation, while the actual output values, which are very valuable, are ignored. In addition, the nonlinear active case data leads to a possible performance degradation of the model in the face of samples that do not appear during the training process due to the lack of generalisation capability. The accuracy obtained by applying it to the COVID-19 active case prediction does not satisfy the real situation. Therefore, we propose here to use the gravitational search algorithm to search for the internal network parameters that are most suitable for ELM to predict the number of active cases of COVID-19, thus improving the overall performance of the model. The overall architecture of the GSA-ELM hybrid

model is shown in Figure 2. As can be seen, we first initialise the parameters of the GSA and the particle positions of the population and evaluate the fitness value of each particle. Next, we calculate the interaction forces between the particles to update the velocity and position of each particle. The algorithm is iterated several times until the termination condition is satisfied. At this point, the global optimal solution of the problem returned by the GSA is applied to the parameter settings of the ELM. Finally, the ELM is used to generate the final prediction of our model for the number of future active cases. Next we present the details of the gravitational search algorithm as follows.

In 2009, Rashedi et al. [27] proposed a novel optimisation algorithm, the Gravitational Search Algorithm (GSA), based on the law of gravity and the interaction between particles. The mass of the individual measures the merit of the individual, and the better the position, the greater the mass. Due to the effect of gravity, individuals are attracted to each other and move in the direction of the individual with the largest quality. As the movement continues, ultimately the whole group will gather around the individual with the largest mass, thereby finding the individual with the largest mass, which occupies the best

position. Hence, the algorithm is able to obtain the optimal solution to the problem. In the model, we consider a search space with N particles. We define the position of the r th particle as follows:

$$\mathcal{M}_r = (\mu_r^1, \dots, \mu_r^d, \dots, \mu_r^D), r = 1, 2, \dots, N, \quad (4)$$

where μ_r^d represents the position of the r th particle in the d th dimension. The dimension of each particle is $D = k(n + 1)$, k and n are the number of neurons in the hidden and input layers in ELM, respectively. Similarly we define the velocity of the r th particle among N particles as:

$$\mathcal{V}_r = (v_r^1, \dots, v_r^d, \dots, v_r^D), r = 1, 2, \dots, N, \quad (5)$$

where v_r^d denotes the velocity of the r th particle in the d th dimension.

Based on the principle of force interaction, in the i th iteration we define the force acting on the r th particle by the s th particle as follows:

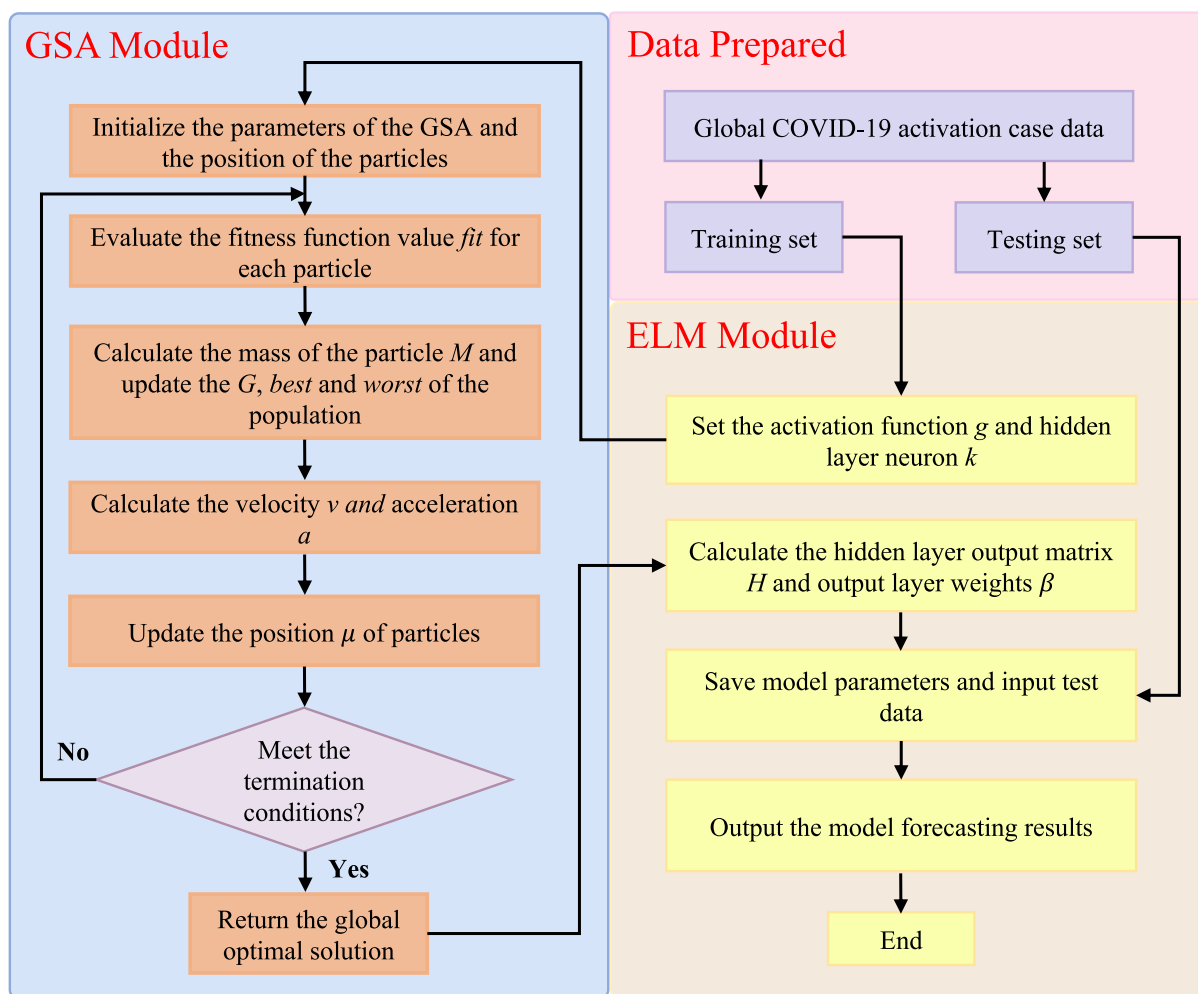


FIGURE 2 The general structure of our proposed GSA-ELM model for COVID-19 active cases forecasting.

$$F_{rs}^d = G(i) \frac{M_r(i) \times M_s(i)}{R_{rs}(i) + \sigma} (\mu_s^d(i) - \mu_r^d(i)), \quad (6)$$

where $R_{rs}(i)$ denotes the Euclidean distance between the particles r and s . M_r and M_s are respectively the passive gravitational mass of particle r and the active gravitational mass of particle s . σ is a small constant in order to prevent the denominator turning into zero. $G(i)$ denotes the gravitational constant, whose value decreases with each iteration of the algorithm to control the search accuracy. The specific representation of $G(i)$ is as follows:

$$G(i) = G_0 e^{-\epsilon(i)}, \quad (7)$$

G_0 is the initial value of the gravitational constant $G(i)$, I is the overall number of iterations of the algorithm and ϵ is the constant which requires adjustment.

To introduce the complex property in the algorithm, assume that the total force acting on the r th particle in the d th dimension is a random weighted sum of the forces exerted by the other particles, expressed as:

$$F_r^d = \sum_{s=1, s \neq r}^N \text{rand}_s F_{rs}^d(i), \quad (8)$$

where $\text{rand}_s \in (0, 1)$ represents a random variable that follows a uniform distribution, and to add the random property.

The algorithm must start with a full exploration of the search space to prevent falling into a local optimum. As the iterations proceed, the search gradually decreases and the exploitation gradually increases. Also, to improve the performance of GSA by controlling the exploration and exploitation, a good compromise is to reduce the number of particles over time. Specifically, only δ_{best} optimal particles can attract other particles, where δ_{best} is a function of time and decreases linearly with the iteration process. Finally only one particle applies force on the other particles. Therefore, Equation 8 can be modified as follows:

$$F_r^d = \sum_{s \neq r, s \in \delta_{best}}^N \text{rand}_s F_{rs}^d(i), \quad (9)$$

where δ_{best} is the set of the first δ particles has the largest mass and the best fitness value. According to Newton's second law, the acceleration of particle r at the i th iteration and in the d th dimension, is defined as:

$$a_r^d(i) = \frac{F_r^d(i)}{M_r(i)}, \quad (10)$$

where M_r represents the inertial mass of r th particle. The next velocity of particle r is considered to be part of the current velocity plus its acceleration, which leads to the calculation of its velocity and position update by:

$$v_r^d(i+1) = \text{rand}_r \times v_r^d(i) + a_r^d(i), \quad (11)$$

$$\mu_r^d(i+1) = \mu_r^d(i) + v_r^d(i+1), \quad (12)$$

where $v_r^d(i+1)$ and $\mu_r^d(i+1)$ indicate the velocity and position of particle r after the $(i+1)$ th iteration update, respectively. $\text{rand}_r \in (0, 1)$ is also a random variable obeying uniform distribution.

The mass of the individual is obtained by fitness assessment, the greater the mass the better the position, the stronger the attraction, and the slower the movement. The inertial mass and gravitational of particle r are updated according to the following equation:

$$m_r(i) = \frac{\text{fit}_r(i) - \text{worst}(i)}{\text{best}(i) - \text{worst}(i)}, \quad (13)$$

$$M_r(i) = \frac{m_r(i)}{\sum_{s=1}^N m_s(i)}, \quad (14)$$

where $\text{fit}_r(i)$ and $M_r(i)$ respectively represent the value of the MAPE and mass of the r th particle in the i th iteration. The specific definitions of $\text{best}(i)$ and $\text{worst}(i)$ are as follows:

$$\text{best}(i) = \min_{s \in \{1, 2, \dots, N\}} \text{fit}_s(i), \quad (15)$$

$$\text{worst}(i) = \max_{s \in \{1, 2, \dots, N\}} \text{fit}_s(i), \quad (16)$$

the $\text{best}(i)$ and $\text{worst}(i)$ represent the best fitness function value and the worst fitness function value among all particles at the i th iteration.

4 | CASE STUDY

The data utilised in this section are from the data repository of the 2019 Novel Coronavirus Visual Dashboard operated by the Johns Hopkins University Centre for Systems Science and Engineering (JHU CSSE).¹ In the second subsection a brief description of the criteria for experimental evaluation is provided. Our experiments were conducted on a computing hardware environment with Intel Core i7 3.60 GHz and 8 GB RAM, running on Python 3.7.

4.1 | Data description

The active cases for the case study were obtained from the data repository of the 2019 Novel Coronavirus Visual Dashboard operated by the Johns Hopkins University Centre for Systems Science and Engineering (JHU CSSE), which includes daily case reports for COVID-19 worldwide, daily status reports for the United States, and time series summaries. The timestamped

¹<https://github.com/CSSEGISandData/COVID-19>

of all cases are denoted in UTC (GMT + 0). The number of active cases for each country and region are aggregated by each country and region from January 27, 2020 to December 21, 2020.

4.2 | Evaluation criteria

This experiment adopts two common evaluation metrics to assess the performance of our model. These two criteria are the mean absolute percentage error (MAPE) and the root mean square error (RMSE), which are defined as follows.

$$RMSE = \sqrt{\frac{1}{S} \sum_{s=1}^S (\hat{f}_s - f_s)^2}, \quad (17)$$

$$MAPE = \frac{1}{S} \sum_{s=1}^S \left| \frac{\hat{f}_s - f_s}{f_s} \right|, \quad (18)$$

where S is the number of test samples, f_s and \hat{f}_s are the true measured and predicted values of the s th test sample.

4.3 | Experiment & results analysis

The projections for COVID-19 active cases are intended to anticipate fluctuations in the population over time, rather than to anticipate variations in the population on a per-day basis. Hence we calculated the number of active cases for each country and region by adding up the original data to obtain the total number of active cases for that day as one of our data samples. We selected each experimental sample to be sequenced by time series and intercepted the data using a rolling method with a rolling step set to 1. The data were counted every 5 days as one group, resulting in a total of 326 sets of data. The data sample for each set is divided into two parts, the first part consists of the data from the first 4 days and the second part consists of the data from the last day, which is the value to be predicted, that is, the target value. Hence, the ELM network has a 4-dimensional input vector and a 1-dimensional output vector. The overall data is also divided into two parts, where 80% of the data is utilised to train our model, while the residuals are used to test the performance of the model. In this study, two evaluation criteria, RMSE and MAPE, are applied to gauge the performance of GSA for ELM optimisation. In purpose to have a better training effectiveness of the ELM model, we set the number of neurons in the hidden layer within the network to 100, while the specific settings of the GSA parameters are shown in Table 1. In our experiments we also analysed the effect of different number of iterations on the prediction performance of the model by setting the number of iterations of the model to different values (1–100). We show in Figure 3 the results comparing MAPE for different number of iterations. We can observe that when the number of iterations exceeds 30, the particles in the GSA move to the optimal position and MAPE

TABLE 1 The GSA parameters setting.

Parameter	Value
Gravitational constants initial value G_0	100
Constant ϵ	20
Number of iterations I	100
Number of particles N	40
Constant σ	1

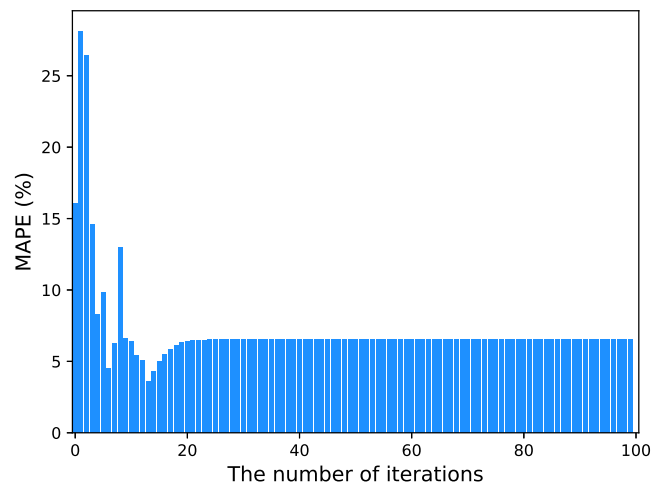


FIGURE 3 MAPE of the prediction results for different number of iterations. After the number of iterations exceeds 30, the MAPE value tends towards stability.

values converge to a smooth value. At this point, the performance of the model does not improve significantly after more iterations. Therefore, in this experiment we set the initial number of iterations of the model to 100 to ensure that the model can fully converge and that the model will not lose accuracy due to insufficient training.

We also compare the performance of our model with a model that utilises the particle swarm algorithm (PSO) to optimise the ELM to demonstrate the superiority of GSA in determining the ELM parameters. For the purpose of comparison between algorithms, all parameters should be considered as the same criteria, except for algorithm parameters that require special settings. The particular parameter settings in PSO follow previous work in our experiments [52, 19]. Specifically, we set the social learning factor c_1 and the individual learning factor c_2 in the algorithm to 0.55 and 0.35, respectively, and the inertia weight ω that regulates the search range over the solution space is set to the default 0.9. The number of particle swarms is 20 and the maximum number of iterations is likewise 100. Table 2 shows the prediction results of the ELM model with optimised ELM parameters using the GSA and PSO algorithms, respectively, versus the ELM model without any optimisation algorithm. We can see that optimising the parameters of the ELM by both the GSA and PSO algorithms can effectively improve the performance of the ELM. Compared with the ordinary ELM, GSA helps the ELM network solve the optimal

TABLE 2 Comparison of forecasting performance and computational time overhead for different ELM-based models.

Method	MAPE	RMSE	Training (s)	Testing (s)
ELM [18]	7.09	831957.76	0.033	1.9×10^{-3}
PSO-ELM	5.81	498020.46	1039.87	2.9×10^{-3}
GSA-ELM	3.64	397025.31	482.53	2.4×10^{-3}

Note: Bold values denote the best one among that column.

input weight vector and bias value parameters that are most suitable for the prediction task by simulating the motion of particles. Not only can it effectively help ELM to get rid of local extremes and thus obtain optimal results, but also the optimised model can have certain generalisation ability. In particular, the GSA-ELM model proposed in this paper has obvious advantages over the PSO-ELM model in both metrics. The PSO algorithm is easy to fall into the local extrema for functions with multiple local extrema in the optimisation problem, thus obtaining suboptimal results. Meanwhile, the PSO method offers the possibility of global search, but does not strictly prove its convergence on the global optimal point. The number of active cases of COVID-19 shows a nonlinear variation with time, and the parameter optimisation problem of ELM has multiple local extremum points, so it does not perform better compared to the GSA algorithm.

We also report the time overhead of the three models for training and testing. In the training phase, the ELM models based on the optimisation algorithm all take more time. Compared to the ELM, it just randomly assigns parameters without the optimisation process and is relatively less accurate. Secondly, GSA has faster convergence speed and superior prediction effect than PSO, which is more practical in the actual online active cases prediction. And in the testing phase, the time overheads of the three models are roughly similar, the reason being that the inference of the final prediction results all depend mainly on the speed of ELM.

4.4 | Performance evaluation

Active cases prediction was performed utilising several conventional machine learning models, where the datasets and data are divided in a similar way as before for the comparison experiments. These models were implemented by invoking the sklearn algorithm library, and the parameters of the models mostly used the default settings in sklearn. Table 3 exhibits the prediction performance of each model. The individual models are described below.

KNN [56]: Through finding the k nearest neighbors of a sample and assigning the average of some attribute(s) of these neighbors to that sample, the value of the corresponding attribute(s) of that sample can be obtained. The choice of k value in KNN algorithm will have a large impact on the prediction performance of the model. If a smaller value of k is chosen, it is equivalent to predicting with training instances in a smaller domain, which means that the overall model becomes

TABLE 3 The forecasting results of GSA-ELM and other contrast models.

Method	MAPE	RMSE	Training (s)	Testing (s)
SVR [53]	12.54	1156809.46	0.039	5.3×10^{-3}
DT [54]	13.72	1242751.27	0.019	3.1×10^{-3}
KF [55]	12.95	1204367.18	0.154	7.3×10^{-3}
KNN [56]	10.47	845803.09	0.011	1.6×10^{-2}
ANN [57]	10.76	813295.43	6.42	2.8×10^{-3}
Ridge Regression [58]	8.94	705551.56	0.115	4.0×10^{-3}
ELM [18]	10.19	595782.92	0.033	1.9×10^{-3}
GSA-ELM	7.79	474522.09	482.53	2.4×10^{-3}

Note: Bold values denote the best one among that column.

complex and prone to overfitting; if a larger value of k is chosen, it is equivalent to predicting with training instances in a larger domain, which has the advantage of reducing the estimation error of learning, but the disadvantage that the approximation error of learning increases. Therefore, considering the above reasons, we experimentally set the value of k to 4 to make the KNN model have better prediction performance and robustness to noise.

DecisionTree (DT) [54]: The study adopts a Decision Tree model based on Classification and Regression Tree (CART) for prediction. CART does not require any a priori assumptions and is highly resistant to noise and missing data. Set where the max_length parameter is 5.

SVR [53]: The principle of SVR is to locate a regression plane in which all data of a collection have the closest distance to that plane. This experiment applies the best-fitted Gaussian RBF kernel function.

RidgeRegression [58]: A regularised version of linear regression, is a biased estimation regression method, which is essentially a modified least squares estimation method. For which the regularisation parameter α is set to 0.5.

ANN [57]: The artificial neural network can continuously learn to extract the features of each part of the data, and change the strength of each connection by training the network weights of the connections until the output of the top layer gets the correct answer. We set the number of hidden layers here to 1 and the number of nodes to 50.

KF [55]: Kalman Filtering (KF) provides optimal estimation of the system state through the system input and output observations. We set the variance of the process error Q to $0.1 \times I$, where I represents the identity matrix. We set the variance of the measurement noise to 0. The covariance matrix of the initial state estimation error is denoted as $10^{-2} \times I$.

Among the GSA algorithm and PSO algorithm, the particles are randomly distributed and the results of each experiment are different. Therefore, the experimental results of both the GSA-ELM model and the PSO-ELM model are run over 100 times. Multiple experiments were conducted for each model in the comparison experiments and the final results of

each outcome were averaged to ensure the fairness of our experiments. The comparison of the prediction performance and time overhead between the different methods is shown in Table 3. We can observe that: 1) GSA-ELM has better performance in both metrics compared to other traditional machine learning methods. For example, the MAPE value decreases by 14.8% compared to the ridge regression model, which has the smallest MAPE value among the other five models. Since SVR solves support vectors with the help of quadratic programming, and solving quadratic programming will involve the computation of a matrix of order m (m is the number of samples), the storage and computation of this matrix will consume a lot of machine memory and computing time when the number of m is large. At the same time, the performance of regression mainly depends on the selection of kernel function, so the actual problem of COVID-19 active case prediction, how to choose the appropriate kernel function according to the actual data model to construct the SVR algorithm is still very challenging. The Kalman Filter does not achieve optimal estimation in the nonlinear scenario of the COVID-19 active case because it only provides accurate estimation of the linear process and measurement model. KNN prediction results are easily affected by noisy data, the number of active cases of COVID-19 is not stable from day to day, and the category of new samples biased towards the category with the dominant number in the training sample, which easily leads to prediction errors. It also has high computational complexity and memory consumption because for each text to be classified, the distance to all known samples has to be calculated to find its K nearest neighbors, and the computational time overhead is also long. The Ridge Regression method is essentially a modified least squares estimation method that requires a more realistic and reliable regression coefficient at the cost of losing some information and reducing accuracy by abandoning the unbiased nature of the least squares method. The ELM has good generalisation performance and remains highly robust to the number of COVID-19 active cases affected by various background noises. Meanwhile, the RMSE value decreased by 25.6% compared with the ELM model with the smallest RMSE, which proved the effectiveness of the GSA global optimisation of the parameters of the ELM. 2) GSA-ELM relies on the continuous shifting of the position of the particles of the population to find the global optimal solution of the problem. Therefore, it will take longer time in the model training phase compared to other methods with lower complexity. However, when comparing the time overhead of the testing phase, it is clear that ELM has the fastest speed. This is because in the training phase ELM derives the weights from the hidden layer to the output layer by inverse operations, while GSA determines the optimal input weight values and the bias values of the hidden neurons. In the test phase ELM only needs to perform a simple matrix multiplication operation to accomplish the task of active cases number prediction. The test phase has a similar time cost as other machine learning, which greatly satisfies the need for fast prediction of the number of activated cases in reality.

In addition, in order to better demonstrate the performance of our model and avoid the overlapping of the forecasting result lines under the large scale condition that makes multiple lines indistinguishable. For this reason, we selected several periods of time when the number of active people exploded for presentation. From Figure 4, we can notice that the forecasted value of GSA-ELM basically coincides with the real value, while the other six models have a little discrepancy with the real value, where the green line represents the real value and the red line represents the forecasted value of GSA-ELM. We can observe that the GSA-ELM model has high accuracy and stability in most circumstances, and its performance is better than other conventional machine learning forecasting methods. In summary, our model has more advanced performance and faster prediction response time compared to other methods for future dynamic global active cases forecasting.

It is worth noting that the model proposed in this paper can be deployed online and the model training and active cases prediction tasks can be performed in parallel. First, we train the model with the help of existing data on the number of active cases and deploy it online for prediction work. In this case, the model takes little time to complete the prediction task compared to other methods. Second, the prediction capability is further enhanced by collecting the latest reported daily active cases data worldwide to continuously train and optimise the internal parameters of our model. In addition, the latest trained parameters are uploaded in parallel while running online. Finally, the ELM with updated parameters is used to make more accurate and faster predictions of the dynamic number of COVID-19 active cases in the future. The model can effectively help government agencies and related organisations to make appropriate epidemic prevention decisions quickly and further prevent the spread of the epidemic among the population.

4.5 | Dataset division proportion

We split the overall data into two parts, that is, the training set and the test set for the evaluation of the model's parameter learning and predictive capability, respectively. We therefore analysed the impact of different splitting ratios on the prediction performance here. The detailed results are presented in Table 4. We can see that when the proportion of the training set is less than or equal to 80%, we obtain better results as the proportion of the training set increases. In particular, the RMSE and MAPE decrease by 10.9% and 6.6%, respectively, when the ratio of training set to test set reaches 70%:30%. This is due to the fact that the increase in the training set allows the model to be more adequately trained to learn more complex nonlinear patterns, which is more beneficial to improve the performance and robustness of the model in the face of data noise. We can also see that the metrics do not improve significantly when the proportion of the training set reaches

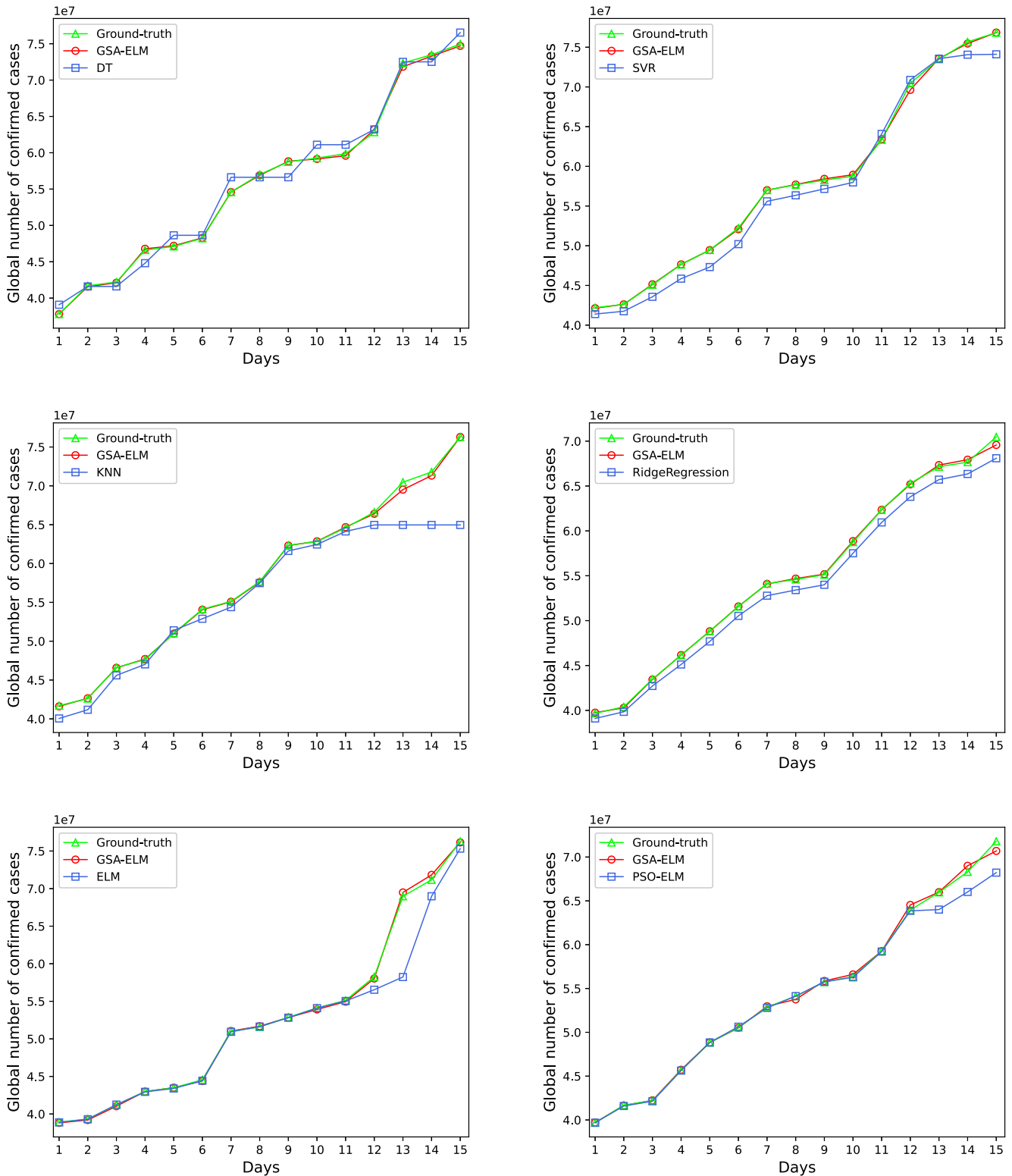


FIGURE 4 Six examples demonstrate our model outperforms the state-of-the-art models.

90%. A large proportion of the training set may result in training a model closer to the one trained with the total sample, increasing the possibility of data leakage. Also the model is prone to overfitting, so it appears to perform poorly on the emerging test set.

4.6 | Analysis of variance (ANOVA) test

To evidence whether there is a significant difference between our model and the other methods, we performed an ANOVA test. Table 5 shows the results. It is generally accepted that a

TABLE 4 The results of using different division proportions of the training and testing sets.

Metric	Training set: Testing set			
	60% : 40%	70% : 30%	80% : 20%	90% : 10%
MAPE	8.62	8.05	7.79	7.87
RMSE	689373.51	614400.75	474522.09	503040.32

Note: Bold values denote the best one among that column.

TABLE 5 Compare the p -values of the different methods in the ANOVA test.

Method	SVR	DT	KF	KNN	RidgeRegression	ELM
p -value	8.45×10^{-5}	1.91×10^{-6}	1.34×10^{-6}	3.13×10^{-5}	4.20×10^{-4}	3.96×10^{-4}

p -value less than 0.05 means that the difference between the two models is statistically significant. We can see that all p -values in the table are less than 0.05, which reflects that our model is more statistically significant compared to other methods.

5 | DISCUSSION

To solve the problem of suboptimal prediction due to random initial internal network parameters of the extreme learning machine. In this paper, a gravitational search algorithm is used to search for the optimal solution to the COVID-19 active case prediction task by simulating the gravitational motion of a swarm of particles. This effectively prevents ELM from falling into local optimality and improves its prediction ability in the face of nonlinear and unstable data. The experimental findings also show that our approach can improve the robustness of the model to a certain extent while keeping its running time overhead. The gravitational search algorithm has some advantages over traditional optimisation algorithms in terms of efficiency in solving nonlinear functions and in solving high-dimensional search space optimisation problems. It also has good search performance compared to other optimisation models. However, only the current position information plays a role in the iterative process, which indicates that the gravitational search algorithm is a method lacking in memory, and there is also the possibility of falling into a local optimum. This difficulty is a problem that optimisation search methods often encounter. This leads to the fact that using it together with ELM may produce predictions that are not very satisfactory. Therefore, in order to break the above limitations, we consider that other optimisation search models can be used to solve or other optimisation algorithms can also be used to optimise the initial parameter values of GSA in order to speed up the overall movement of the population and induce the algorithm to have a stronger search capability.

Currently, there are several studies in the literature that focus on optimising the network model to improve the immunity of the model to noise. For example, Cui et al [59] proposed a two-stage hybrid learning model to search the initial parameter values of the GSA in a data-driven manner with the PSO algorithm to improve the efficiency of the global optimum search. Yin et al [60] introduced a modified GSA with crossover (CROGSA), where the crossover-based search scheme utilises the promising knowledge extracted from the

currently obtained global optimum positions to improve the exploitation capabilities. We have been working on collecting more COVID-19 related data for inclusion in the learning and training of the GSA-ELM model to improve the generalisation capability of the model for deployment in real applications. This will greatly assist the work of the outbreak prevention and control authorities and facilitate the implementation of targeted outbreak prevention and control policies.

6 | CONCLUSION

In this paper, we propose a hybrid learning model GSA-ELM for COVID-19 global active cases forecasting. To predict the complex and multifactorial COVID-19 active cases, we use a gravitational search algorithm to search the global optimal parameters of the extreme learning machine. The experimental results demonstrate the reliability of the model in predicting real-life active cases compared to other state-of-the-art methods, and its excellent generalisation ability is of good application. In addition, our model can be deployed and applied online, the model uses a data-driven approach for training and the prediction of the number of active cases can be performed in parallel. In the future, the model will be able to assist the government and related organisations to make appropriate epidemic prevention plans as early as possible to control the spread of the epidemic among the population and protect people's lives.

AUTHOR CONTRIBUTION

Boyu Huang: Methodology; Software. **Youyi Song:** Formal analysis. **Zhihan Cui:** Validation. **Haowen Dou:** Data curation. **Dazhi Jiang:** Writing – review & editing. **Teng Zhou:** Funding acquisition; Project administration. **Jing Qin:** Supervision.

ACKNOWLEDGEMENT

This work was supported by the National Natural Science Foundation of China (No. 61902232), the 2022 Guangdong Basic and Applied Basic Research Foundation (No. 2022A1515011590), the Project of Strategic Importance of The Hong Kong Polytechnic University (No. 1-ZE2Q), and the 2020 Li Ka Shing Foundation Cross-Disciplinary Research Grant (No. 2020LKSG05D). All authors of this paper would like to thank Mrs. Zhizhe Lin, Ph.D. candidate at Hainan University, for her help with this paper.

CONFLICT OF INTEREST STATEMENT

The authors declare that they have no conflict of interest.

DATA AVAILABILITY STATEMENT

The dataset and source code generated during and/or analysed during the current study are available from the corresponding author upon reasonable request.

ORCID

Boyu Huang  <https://orcid.org/0000-0001-8262-7218>

Teng Zhou  <https://orcid.org/0000-0003-1920-8891>

REFERENCES

- Li, Q., et al.: Early transmission dynamics in wuhan, China, of novel coronavirus-infected pneumonia. *N. Engl. J. Med.* 382(13), 1199–1207 (2020). <https://doi.org/10.1056/nejmoa2001316>
- Wu, Y.-C., Chen, C.-S., Chan, Y.-J.: The outbreak of covid-19: an overview. *J. Chin. Med. Assoc.* 83(3), 217–220 (2020). <https://doi.org/10.1097/jcma.0000000000000270>
- Lu, H., Stratton, C.W., Tang, Y.-W.: Outbreak of pneumonia of unknown etiology in wuhan, China: the mystery and the miracle. *J. Med. Virol.* 92(4), 401–402 (2020). <https://doi.org/10.1002/jmv.25678>
- Zhou, P., et al.: A pneumonia outbreak associated with a new coronavirus of probable bat origin. *Nature (Lond.)* 579(7798), 270–273 (2020). <https://doi.org/10.1038/s41586-020-2012-7>
- Sohrabi, C., et al.: World health organization declares global emergency: a review of the 2019 novel coronavirus (covid-19). *Int. J. Surg.* 76, 71–76 (2020). <https://doi.org/10.1016/j.ijsu.2020.02.034>
- Sajadi, M.M., et al.: Temperature and Latitude Analysis to Predict Potential Spread and Seasonality for Covid-19. Available at SSRN 3550308
- Gandhi, M., Yokoe, D.S., Havlir, D.V.: Asymptomatic Transmission, the Achilles' Heel of Current Strategies to Control Covid-19 (2020)
- Huang, C., et al.: Clinical features of patients infected with 2019 novel coronavirus in wuhan, China. *Lancet* 395(10223), 497–506 (2020). [https://doi.org/10.1016/s0140-6736\(20\)30183-5](https://doi.org/10.1016/s0140-6736(20)30183-5)
- Chan, J.F.-W., et al.: A familial cluster of pneumonia associated with the 2019 novel coronavirus indicating person-to-person transmission: a study of a family cluster. *Lancet* 395(10223), 514–523 (2020). [https://doi.org/10.1016/s0140-6736\(20\)30154-9](https://doi.org/10.1016/s0140-6736(20)30154-9)
- Filatov, A., et al.: Neurological complications of coronavirus disease (covid-19): encephalopathy. *Cureus* 12(3)
- Li, K., et al.: The clinical and chest ct features associated with severe and critical covid-19 pneumonia. *Invest. Radiol.* 55(6), 327–331 (2020). <https://doi.org/10.1097/rli.0000000000000672>
- Poyiadji, N., et al.: Covid-19-associated acute hemorrhagic necrotizing encephalopathy: ct and mri features. *Radiology*, 201187 (2020)
- Fang, W., et al.: Deep learning anti-fraud model for internet loan: where we are going. *IEEE Access* 9, 9777–9784 (2021). <https://doi.org/10.1109/ACCESS.2021.3051079>
- Fang, W., et al.: Attention meets long short-term memory: a deep learning network for traffic flow forecasting. *Phys. Stat. Mech. Appl.* 587, 126485 (2022). <https://doi.org/10.1016/j.physa.2021.126485>
- Huang, H., et al.: Voxel-based morphometry and a deep learning model for the diagnosis of early alzheimer's disease based on cerebral gray matter changes. *Cerebral Cortex* bhac099. arXiv <https://academic.oup.com/cercor/advance-article-pdf/doi/10.1093%20/cercor/bhac099/42950643/bhac099.pdf>
- Dou, H., et al.: Transfer inhibitory potency prediction to binary classification: a model only needs a small training set. *Comput. Methods Progr. Biomed.* <https://doi.org/10.1016/j.cmpb.2022.106633>
- Lu, H., et al.: A temporal-aware lstm enhanced by loss-switch mechanism for traffic flow forecasting. *Neurocomputing* 427, 169–178 (2021). <https://doi.org/10.1016/j.neucom.2020.11.026>
- Huang, G.-B., Zhu, Q.-Y., Siew, C.-K.: Extreme learning machine: theory and applications. *Neurocomputing* 70(1–3), 489–501 (2006). <https://doi.org/10.1016/j.neucom.2005.12.126>
- Cai, W., et al.: A hybrid learning model for short-term traffic flow forecasting. *IEEE Access* 8, 6505–6514 (2020). <https://doi.org/10.1109/ACCESS.2019.2963784>
- Cui, Z., et al.: Gsa-elm: a hybrid learning model for short-term traffic flow forecasting. *IET Intell. Transp. Syst.* 16(1), 41–52 (2022). <https://doi.org/10.1049/itr2.12127>
- Wan, C., et al.: Probabilistic forecasting of wind power generation using extreme learning machine. *IEEE Trans. Power Syst.* 29(3), 1033–1044 (2013). <https://doi.org/10.1109/tpwrs.2013.2287871>
- Chen, X., et al.: Electricity price forecasting with extreme learning machine and bootstrapping. *IEEE Trans. Power Syst.* 27(4), 2055–2062 (2012). <https://doi.org/10.1109/tpwrs.2012.2190627>
- Yuan, Y., et al.: Noise-immune extreme ensemble learning for early diagnosis of neuropsychiatric systemic lupus erythematosus. *IEEE journal of biomedical and health informatics* doi 26(7), 3495–3506 (2022). <https://doi.org/10.1109/JBHI.2022.3164937>
- You, C.-X., Huang, J.-Q., Lu, F.: Recursive reduced kernel based extreme learning machine for aero-engine fault pattern recognition. *Neurocomputing* 214, 1038–1045 (2016). <https://doi.org/10.1016/j.neucom.2016.06.069>
- Tang, J., Deng, C., Huang, G.-B.: Extreme learning machine for multi-layer perceptron. *IEEE Transact. Neural Networks Learn. Syst.* 27(4), 809–821 (2015). <https://doi.org/10.1109/tnnls.2015.2424995>
- Huang, G.-B., Chen, L.: Convex incremental extreme learning machine. *Neurocomputing* 70(16–18), 3056–3062 (2007). <https://doi.org/10.1016/j.neucom.2007.02.009>
- Rashedi, E., Nezamabadi-Pour, H., Saryazdi, S.: Gsa: a gravitational search algorithm. *Inf. Sci.* 179(13), 2232–2248 (2009). <https://doi.org/10.1016/j.ins.2009.03.004>
- Chen, N., et al.: Epidemiological and clinical characteristics of 99 cases of 2019 novel coronavirus pneumonia in wuhan, China: a descriptive study. *Lancet* 395(10223), 507–513 (2020). [https://doi.org/10.1016/s0140-6736\(20\)30211-7](https://doi.org/10.1016/s0140-6736(20)30211-7)
- Wang, D., et al.: Clinical characteristics of 138 hospitalized patients with 2019 novel coronavirus-infected pneumonia in wuhan, China. *JAMA* 323(11), 1061–1069 (2020). <https://doi.org/10.1001/jama.2020.1585>
- Patel, A., Jernigan, D.B., 2019-nCoV CDC Response Team: Initial public health response and interim clinical guidance for the 2019 novel coronavirus outbreak—United States, december 31, 2019–february 4, 2020. *MMWR (Morb. Mortal. Wkly. Rep.)* 69(5), 140–895 (2020). <https://doi.org/10.1111/ajt.15805>
- Chen, Y., Liu, Q., Guo, D.: Emerging coronaviruses: genome structure, replication, and pathogenesis. *J. Med. Virol.* 92(4), 418–423 (2020). <https://doi.org/10.1002/jmv.25681>
- Anastassopoulou, C., et al.: Data-based analysis, modelling and forecasting of the covid-19 outbreak. *PLoS One* 15(3), e0230405 (2020). <https://doi.org/10.1371/journal.pone.0230405>
- van den Driessche, P., Watmough, J.: A simple sis epidemic model with a backward bifurcation. *J. Math. Biol.* 40(6), 525–540 (2000). <https://doi.org/10.1007/s002850000032>
- Liu, J., et al.: A networked sis disease dynamics model with a waterborne pathogen. In: 2019 American Control Conference, pp. 2735–2740. ACC, IEEE (2019)
- Cai, Y., Kang, Y., Wang, W.: A stochastic sirs epidemic model with nonlinear incidence rate. *Appl. Math. Comput.* 305, 221–240 (2017). <https://doi.org/10.1016/j.amc.2017.02.003>
- Mojeeb, A., Adu, I.K., Yang, C.: A simple seir mathematical model of malaria transmission. *Asian Research Journal of Mathematics* 7(3), 1–22 (2017). <https://doi.org/10.9734/arjom/2017/37471>
- Almeida, R.: Analysis of a fractional seir model with treatment. *Appl. Math. Lett.* 84, 56–62 (2018). <https://doi.org/10.1016/j.aml.2018.04.015>
- Chen, Y.-C., Lu, P.-E., Chang, C.-S.: A Time-dependent Sir Model for Covid-19. arXiv preprint arXiv:2003.00122
- Tang, B., et al.: Estimation of the transmission risk of the 2019-ncov and its implication for public health interventions. *J. Clin. Med.* 9(2), 462 (2020). <https://doi.org/10.3390/jcm9020462>
- Awadallah, M.A., et al.: An enhanced binary rat swarm optimizer based on local-best concepts of pso and collaborative crossover operators for

- feature selection. *Comput. Biol. Med.* 147, 105675 (2022). <https://doi.org/10.1016/j.combiomed.2022.105675>
41. Dhiman, G., et al.: A novel algorithm for global optimization: rat swarm optimizer. *J. Ambient Intell. Hum. Comput.* 12(8), 8457–8482 (2021). <https://doi.org/10.1007/s12652-020-02580-0>
 42. Thawkar, S., et al.: Breast cancer prediction using a hybrid method based on butterfly optimization algorithm and ant lion optimizer. *Comput. Biol. Med.* 139, 104968 (2021). <https://doi.org/10.1016/j.combiomed.2021.104968>
 43. Arora, S., Singh, S.: Butterfly optimization algorithm: a novel approach for global optimization. *Soft Comput.* 23(3), 715–734 (2019). <https://doi.org/10.1007/s00500-018-3102-4>
 44. Mirjalili, S.: The ant lion optimizer. *Adv. Eng. Software* 83, 80–98 (2015). <https://doi.org/10.1016/j.advengsoft.2015.01.010>
 45. Mirjalili, S., Lewis, A.: The whale optimization algorithm. *Adv. Eng. Software* 95, 51–67 (2016). <https://doi.org/10.1016/j.advengsoft.2016.01.008>
 46. Chakraborty, S., et al.: Covid-19 x-ray image segmentation by modified whale optimization algorithm with population reduction. *Comput. Biol. Med.* 139, 104984 (2021). <https://doi.org/10.1016/j.combiomed.2021.104984>
 47. Li, S., et al.: Slime mould algorithm: a new method for stochastic optimization. *Future Generat. Comput. Syst.* 111, 300–323 (2020). <https://doi.org/10.1016/j.future.2020.03.055>
 48. Tu, J., et al.: The colony predation algorithm. *JBE* 18(3), 674–710 (2021). <https://doi.org/10.1007/s42235-021-0050-y>
 49. Heidari, A.A., et al.: Harris hawks optimization: algorithm and applications. *Future Generat. Comput. Syst.* 97, 849–872 (2019). <https://doi.org/10.1016/j.future.2019.02.028>
 50. Figueiredo, E.M., Ludermitz, T.B.: Effect of the pso topologies on the performance of the pso-elm. In: 2012 Brazilian Symposium on Neural Networks, IEEE, 2012, pp. 178–183
 51. Huang, G.-B., Zhu, Q.-Y., Siew, C.-K.: Extreme learning machine: theory and applications. *Neurocomputing* 70(1-3), 489–501 (2006). <https://doi.org/10.1016/j.neucom.2005.12.126>
 52. Cai, L., et al.: Svrgsa: a hybrid learning based model for short-term traffic flow forecasting. *IET Intell. Transp. Syst.* 13(9), 1348–1355 (2019). <https://doi.org/10.1049/iet-its.2018.5315>
 53. Smola, A.J., Schölkopf, B.: A tutorial on support vector regression. *Stat. Comput.* 14(3), 199–222 (2004). <https://doi.org/10.1023/b:stco.0000035301.49549.88>
 54. Breiman, L., et al.: *Classification and Regression Trees*. Routledge (2017)
 55. Grewal, M.S., Andrews, A.P., Bartone, C.G.: *Kalman Filtering*
 56. Cover, T., Hart, P.: Nearest neighbor pattern classification. *IEEE Trans. Inf. Theor.* 13(1), 21–27 (1967). <https://doi.org/10.1109/tit.1967.1053964>
 57. Krogh, A.: What are artificial neural networks? *Nat. Biotechnol.* 26(2), 195–197 (2008). <https://doi.org/10.1038/nbt1386>
 58. Hoerl, A.E., Kennard, R.W.: Ridge regression: applications to non-orthogonal problems. *Technometrics* 12(1), 69–82 (1970). <https://doi.org/10.1080/00401706.1970.10488635>
 59. Cui, Z., et al.: A two-stage hybrid extreme learning model for short-term traffic flow forecasting. *Mathematics* 10(12), 2087 (2022). <https://doi.org/10.3390/math10122087>
 60. Yin, B., et al.: Improved Gravitational Search Algorithm with Crossover. *Computers & Electrical Engineering* (2017). S004579061731621X

How to cite this article: Huang, B., et al.: Gravitational search algorithm-extreme learning machine for COVID-19 active cases forecasting. *IET Soft.* 17(4), 554–565 (2023). <https://doi.org/10.1049/sfw2.12139>