



# ML-based group method of data handling: an improvement on the conventional GMDH

Mehdi Amiri<sup>1</sup> · Seyfollah Soleimani<sup>1</sup>

Received: 2 April 2021 / Accepted: 19 July 2021 / Published online: 31 July 2021  
© The Author(s) 2021

## Abstract

Machine learning (ML) has been recognized as a feasible and reliable technique for the modeling of multi-parametric datasets. In real applications, there are different relationships with various complexities between sets of inputs and their corresponding outputs. As a result, various models have been developed with different levels of complexity in the input–output relationships. The group method of data handling (GMDH) employs a family of inductive algorithms for computer-based mathematical modeling grounded on a combination of quadratic and higher neurons in a certain number of variable layers. In this method, a vector of input features is mapped to the expected response by creating a multistage nonlinear pattern. Usually, each neuron of the GMDH is considered a quadratic partial function. In this paper, the basic structure of the GMDH technique is adapted by changing the partial functions to enhance the complexity modeling ability. To accomplish this, popular ML models that have shown reasonable function approximation performance, such as support vector regression and random forest, are used, and the basic polynomial functions in the GMDH are replaced by these ML models. The regression feasibility and validity of the ML-based GMDH models are confirmed by computer simulation.

**Keywords** Group method of data handling · Machine learning · ML-based group method of data handling

## Introduction

The group method of data handling (GMDH) was first introduced by Ivakhnenko as a proper approach for detecting nonlinear systems [1]. The GMDH approach employs a family of inductive algorithms for the computer-based mathematical modeling of multiparameter datasets. This method uses fully automatic parametric and structural optimization. The GMDH is a combination of quadratic and higher neurons in a certain number of variable layers that map a vector of input features to the expected response by creating a multistage nonlinear pattern; it is mainly based on decomposition and dominance. In every layer of this network, a different subset of possible combinations in each neuron among the existing features is mapped to the expected response using polynomial functions [2, 3]. Based on the accuracy achieved for

each combination, some weaker combinations are removed in favor of stronger ones. In other words, different layers of the network are configured by reducing the mapping error from the input feature space to the expected response. Like real structures in nature, the GMDH creates a complex combination of relatively simple structures, and each section is adjusted by an evolutionary approach.

GMDH algorithms are characterized by a reasoning method in which sorting is performed on polynomial models that gradually increase in complexity to select the best solution via a specified external criterion. In the basic structure proposed by Ivakhnenko, polynomial mapping functions (mostly quadratic functions) are used in each GMDH neuron and fitted by the least-squares method. A more complex model is configured for mapping from the input space to the output space by multilayer combinations of mapping created by relatively simple polynomial functions. Since it was first developed, several improvements have been proposed for the GMDH. For example, Ohtani et al. [4] used the M-apoptosis concept to propose a neuro-fuzzy GMDH. Kondo [5] changed the basic GMDH structure and replaced the mechanism for using the output of the neuron in the next layer with backpropagation (BP) and feedback

✉ Seyfollah Soleimani  
s-soleimani@araku.ac.ir

Mehdi Amiri  
87amiri@gmail.com

<sup>1</sup> Department of Computer Engineering, Faculty of Engineering, Arak University, 38156-8-8349 Arak, Iran

structures to the input layer. Elattar et al. [6] combined GMDH and local regression to develop a generalized locally weighted GMDH. Moreover, Zhang et al. [7] developed the diversity-GMDH (D-GMDH) model by using the diversity concept in the GMDH to improve the noise-immunity ability. Shahsavari et al. [8] changed the GMDH structure and added initial inputs to the input binary combinations in the next layers to propose a generalized GMDH structure for modeling thermal conductivity. Band et al. [9] added a sigmoid transfer function to the basic polynomial function of the GMDH method, introduced a neural network-based GMDH (GMDH-NN), and tested it for voltage regulation. Zounemat-Kermani and Mahdavi-Meymand [10] developed the GMDH-FA by the automatic tuning of the GMDH with the silkworm moth algorithm and employed it for the aeration modeling of spillways.

Some studies have combined GMDH with other methods for improving its accuracy. For example, reference [11] predicted bridge pier scour depth under debris flow effects by combining the Adaptive Neuro-Fuzzy Inference System (ANFIS) and GMDH and constructing FN-GMDH. Reference [12], proposed a GMDH-based hybrid model to forecast the container throughput. Considering the complexity of forecasting nonlinear subseries, the proposed model adopts three nonlinear single models—namely, support vector regression (SVR), BP neural network, and genetic programming (GP), to predict the nonlinear subseries. Then, the model establishes selective combination forecasting by the GMDH neural network on the nonlinear subseries and obtains its combination forecasting results. Finally, the predictions of the two parts are integrated to obtain the forecasting results of the original container throughput time series. In reference [13], the GMDH network was developed using a gene-expression programming (GEP) algorithm. In this study, GEP was performed in each GMDH neuron instead of the polynomial quadratic neuron. Effective parameters on the three-dimensional scour rates include sediment size, pipeline geometry, and wave characteristics upstream of the pipeline. Four dimensionless parameters were considered input variables by means of the dimensional analysis technique. Furthermore, scour rates along the pipeline, the vertical scour rate, and scour rates to the left and right of the pipeline are determined as output parameters. Reference [14] combined GMDH with ANFIS to combine their abilities in forecasting ultimate pile bearing capacity. In this study, uncertainty in the data is handled using ANFIS, and the complexity of the input–output relationship is considered using GMDH. In addition, Particle Swarm Optimization (PSO) was used to determine the parameters of these methods. Reference [15] developed a novel hybrid intelligent model for solving engineering problems using a new combination of the GMDH algorithm. In this study, the conventional structure of GMDH is combined with new polynomial functions to

form a new version of the GMDH algorithm by combining fuzzy logic theory, GMDH, and a gravitational search algorithm (GSA). The developed model was leveraged to predict rock tensile strength based on experimental datasets. In this method, simple polynomial functions are replaced with fuzzy if–then rules, which are constructed using Gaussian membership functions, and GSA is used for determining parameters of Gaussian membership functions.

In reference [17], the authors designed a special classifier ensemble selection approach called GMDH-PSVM. The presented work proposed taking advantage of GMDH-NN to further increase the classification performance of SVM. One weakness of the symmetric regularity criterion of GMDH-NN is that if one of the input attributes has a relatively large range, it may overcome the other attributes. Thus, authors first define a standardized symmetric regularity criterion (SSRC) to evaluate and select the candidate models and optimize a classifier ensemble selection approach. Second, they define a novel structure of the initial model of GMDH-NN, which is from the posterior probability outputs of SVMs. These probabilistic outputs were generated from the improved version of Platt's probabilistic outputs. Third, in real classification tasks, different classifiers usually have different classification advantages. Reference [18] proposed a novel hybrid wavelet time series decomposer and GMDH-extreme learning machine (ELM) ensemble method called Wavelet-GMDH-ELM (WGE) for workload forecasting, which predicts and ensembles workload in different time–frequency scales. In [19], GMDH and Genetic Algorithm (GA) were integrated to optimize the ability of GMDH. The efficiency and effectiveness of the GMDH network structure were optimized by the GA, enabling each neuron to search for its optimum connections set from the previous layer. With this proposed model, monitoring data, including the shield performance database, disc cutter consumption, geological conditions, and operational parameters, could be analyzed.

Following the aforementioned works, this study aims to enhance the ability of GMDH to handle more complex relationships between inputs and outputs, which has not been considered before. Considering the reasonable results of ML models in different regression and pattern recognition applications [20–30], it is valuable for us to study whether the combination of ML models and GMDH leads to better performance. A modified version of the GMDH is proposed, in which the basic polynomial functions are replaced by ML models. Given the ability of ML models to establish linear and nonlinear mapping, these methods replace the polynomial functions for the mapping from the inputs to the output in each GMDH neuron. Accordingly, the ML-based GMDH aims to find an optimal approximation in the spanned space by layers that consist of neurons in the ML models. Tests confirm the feasibility and validity of the proposed model

in approximation tasks. The main contributions of this paper are as follows:

1. Improving the accuracy of the GMDH model in forecasting more complex relationships.
2. Replacing conventional polynomial functions with ML models to handle complexities in the datasets.

The rest of this paper is organized as follows: the GMDH mechanism is presented in Sect. 2. The ML-based GMDH is introduced in Sect. 3. The simulation experiments are discussed in Sect. 4, and finally, concluding remarks are presented in Sect. 5.

### Group method of data handling (GMDH)

The GMDH is a nature-inspired learning method that approximates the relationship of inputs and the output by a nonlinear mapping composed of successive layers of neurons using polynomial transfer functions. A basic explanation for the mapping problem is to identify a function ( $\hat{f}$ ) as an alternative for a latent utility function ( $f$ ) to predict  $\hat{y}$  from the input  $X = (x_1, x_2, x_3, \dots, x_n)$  to be as close to the expected output ( $y$ ) as possible. To this end,  $M$  observations, including the multivariable unit–single variable output, are considered as follows [1–3, 10, 31]:

$$y_i = f(x_{i1}, x_{i2}, x_{i3}, \dots, x_{in}), \quad (i = 1, 2, 3, \dots, M). \tag{1}$$

The GMDH network is trained by the input vector  $X$  for predicting the  $\hat{y}$  values:

$$\hat{y}_i = \hat{f}(x_{i1}, x_{i2}, x_{i3}, \dots, x_{in}), \quad (i = 1, 2, 3, \dots, M). \tag{2}$$

The main issue is to determine a GMDH model to ensure the minimization of the squares of the difference between the predicted and expected values, as in the following [10]:

$$\sum_{i=1}^M \left[ \hat{f}(x_{i1}, x_{i2}, x_{i3}, \dots, x_{in}) - y_i \right]^2 \Rightarrow \min. \tag{3}$$

The detailed representation of the Volterra functional series may represent the relationship between the inputs and the output by referring to the Kolmogorov–Gabor polynomial. The output is as follows [1, 2, 31]:

$$y = f(x_1, \dots, x_n) = a_0 + \sum_{i=1}^n a_i x_i + \sum_{i=1}^n \sum_{j=1}^n a_{ij} x_i x_j + \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n a_{ijk} x_i x_j x_k + \dots, \tag{4}$$

where,  $a_0$ ,  $a_i$ , and  $a_{ij}$  are coefficients of polynomial functions. The complete form of mapping the modeling in each neuron is simplified by the output obtained from the partial polynomial functions with two variables as inputs (neurons) [1], as shown in the following equation:

$$\hat{y} = G_{pq}(x_p, x_q) = a_0 + a_1 x_p + a_2 x_q + a_3 x_p x_q + a_4 x_p^2 + a_5 x_q^2. \tag{5}$$

In this approach, a recursive polynomial function is applied to the neurons connected to the network to develop the standard relationship between the input and the output in Eq. 4. The  $a_i$  coefficients in Eq. 5 are calculated by regression to reduce the difference between the observed output ( $\hat{y}$ ) and the expected output ( $y$ ) for each pair of inputs ( $x_p, x_q$ ). In other words, a tree set of polynomial functions in Eq. 5 is developed in which its coefficients are calculated by the least-squares method. The coefficients of each polynomial function ( $G_{pq}$ ) are determined to apply the optimal fitting for the output that corresponds to the input–output pair in the dataset [2, 31] as follows:

$$MSE = \frac{\sum_{i=1}^M (y_i - G_{pq}(x_{ip}, x_{iq}))^2}{M} \Rightarrow \min. \tag{6}$$

To avoid overfitting, 70%–80% ( $P_{\text{train}}$ ) of the total of  $M$  observations is used practically for fitting by the least-squares method, and the rest are used as a validation set for evaluating the approximation error. In other words, Eq. 7 is minimized by the least-squares method, and the value calculated by Eq. 8 is considered the error criterion for each neuron [3]:

$$MSE_{\text{train}} = \frac{\sum_{i \in \text{Train}} (y_i - G_{pq}(x_{ip}, x_{iq}))^2}{M}, \Rightarrow \min$$

Train = ( $P_{\text{train}} * M$ ) number of observations selected randomly. (7)

$$MSE_{\text{validation}} = \frac{\sum_{i \in \text{Validation}} (y_i - G_{pq}(x_{ip}, x_{iq}))^2}{M},$$

Validation =  $((1 - P_{\text{train}}) * M)$  number of observations selected randomly. (8)

In the standard GMDH, all the possible binary combinations of the  $n$  input variables are considered for creating the regression polynomial in Eq. 5 to find the best fitting variables using independent observations ( $y_i, i \in \text{Train}$ ) and the least-squares method. Hence,  $\binom{n}{2} = \frac{n(n-2)}{2}$  neurons exist in the first GMDH layer using  $\{(y_i, x_{ip}, x_{iq}); i \in \text{Train}\}$  observations [2, 3, 19], as shown in the following:

$$\begin{bmatrix} x_{1p} & x_{1q} & \vdots & y_1 \\ x_{2p} & x_{2q} & \vdots & y_2 \\ \dots & \dots & \dots & \dots \\ x_{Mp} & x_{Mq} & \vdots & y_M \end{bmatrix}, \tag{9}$$

where,  $p, q \in \{i = 1, 2, \dots n\}$ . The matrix factorization relations are respectively obtained by Eqs. 10–14 by adding the quadratic sub-equations using Eq. 5 for each row of the  $M$  three-member sets  $(y_i, x_{ip}, x_{iq})$ :

$$Aa = Y, \tag{10}$$

$$a = \{a_0, a_1, a_2, a_3, a_4, a_5\}, \tag{11}$$

$$Y = \{y_1, y_2, y_3, \dots, y_5\}^T, \tag{12}$$

$$A = \begin{bmatrix} 1 & x_{1p} & x_{1q} & x_{1p}x_{1q} & x_{1p}^2 & x_{2p}^2 \\ 1 & x_{2p} & x_{2q} & x_{2p}x_{2q} & x_{2p}^2 & x_{2q}^2 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & x_{Mp} & x_{Mq} & x_{Mp}x_{Mq} & x_{Mp}^2 & x_{Mq}^2 \end{bmatrix}. \tag{13}$$

The parameters are obtained as follows, using the least-squares method and the above equations:

$$a = (A^T A)^{-1} A^T Y. \tag{14}$$

The result gives  $a$  in Eq. 5 for all three values of  $(y_i, x_{ip}, x_{iq})$  of the dataset. This process is repeated for all the neurons in the following layers, which are specified by the internal linkage of the GMDH network [1, 3].

To prevent computational overburden, some neurons in each layer of the GMDH are excluded by a natural selection mechanism. By comparing the sum of squares of the fitting errors for each neuron with a threshold, some neurons and their outputs are excluded from the network. The threshold is calculated from the following equation [10]:

$$T_l = \alpha \text{MSE}_{\text{validation}l}^{\min} + (1 - \alpha) \text{MSE}_{\text{validation}l}^{\max}, \tag{15}$$

where  $l$  represents the  $l$ th layer;  $T_l$  is the threshold value; and  $\text{MSE}_{\text{validation}l}^{\min}$  and  $\text{MSE}_{\text{validation}l}^{\max}$  show the minimum and maximum mean square errors (MSEs) of the fitting among neurons of each layer, respectively; and  $\alpha$  is the selection leverage and a regulatory parameter of the GMDH network. The other regulatory parameters include the number of layers and the maximum allowable number of neurons in each layer (another variable for controlling the model complexity) that controls the GMDH complexity. Figure 1 shows the structure of a hypothetical GMDH network with three middle layers and four inputs [32].

In the next section, the developed ML-based GMDH is presented.

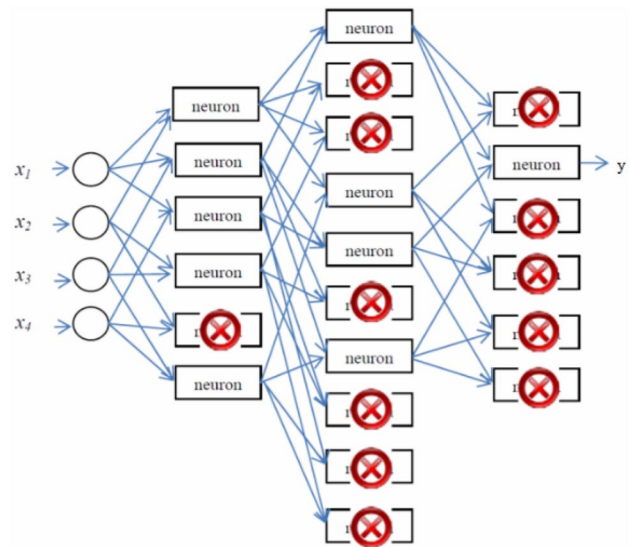


Fig. 1 GMDH with three layers and four inputs [19]

### Machine learning-based group method of data handling (ML-based GMDH)

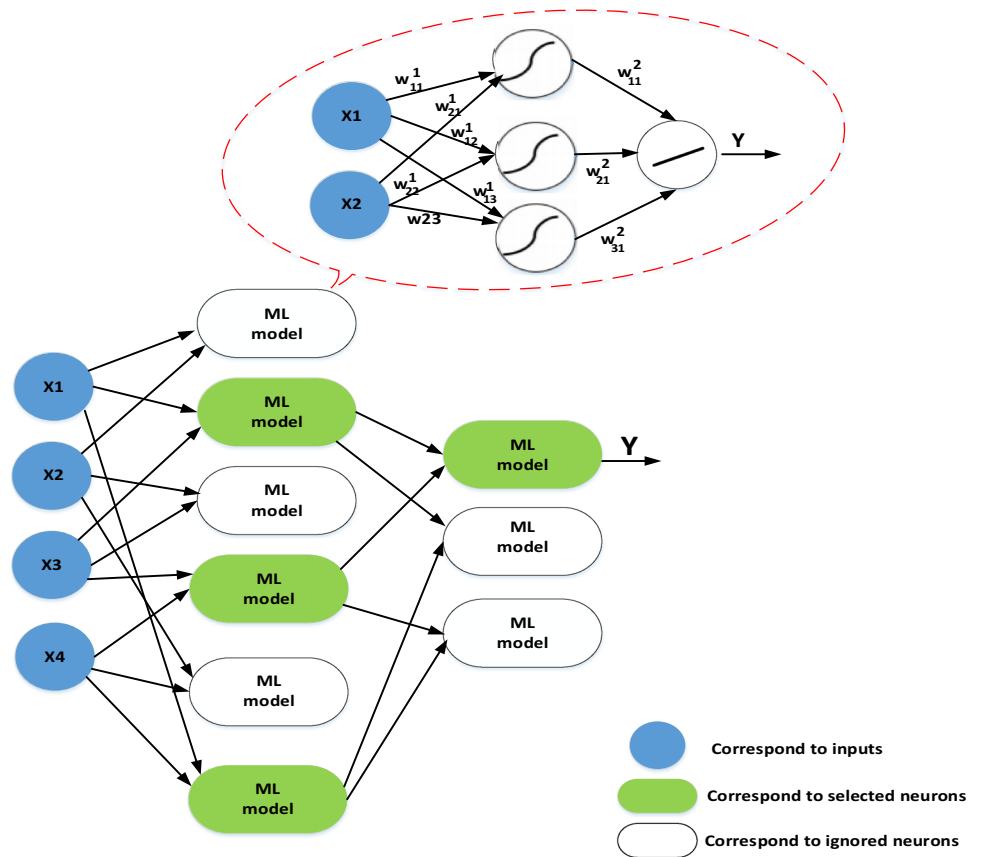
As mentioned in Sect. 2, in each neuron of the GMDH, a polynomial function is fitted between two (or more) inputs and outputs. In the ML-based GMDH models, the partial polynomial functions in the GMDH model are replaced by the ML models. In other words, while preserving the original GMDH structure, the function used for mapping the input pair  $(x_{ip}, x_{jq})$ ,  $i = 1, 2, \dots, M$ , to  $(y_i, i = 1, 2, \dots, M)$  is made more complex than the basic function to model more complex mappings. In cases where there are complex patterns between the input and output pairs, the use of ML models provides GMDH building blocks with more precise approximations.

Unlike the basic GMDH structure in which the  $a$  values in Eq. 5 are obtained by solving Eq. 15, in each neuron of the ML-based GMDH, the ML model is trained once by observations to determine a list of weights and ML parameters. The approximation error for validation observations was calculated by evaluating the outputs of the ML model. Each partial function (neuron) in the ML-based GMDH is considered a black box, and like the basic GMDH structure, the outputs of neurons in each layer are considered inputs to the next layer. The mean square error of validation observations in each neuron of the ML-based GMDH is calculated from the following equation:

$$\text{MSE}_{\text{validation}} = \frac{\sum_{i \in \text{Validation}} (y_i - \text{ML}_{pq}(x_{ip}, x_{iq}))^2}{M},$$

Validation =  $((1 - P_{\text{train}}) * M)$  number of observations selected randomly, (16)

**Fig. 2** ML-based GMDH model with four input variables, two middle layers, and MLP partial functions



where,  $ML_{pq}$  is the trained ML model on  $\{(y_i, x_{ip}, x_{iq}); i \in \text{Train}\}$  observations. Like the basic GMDH model and the selection mechanism of neurons in each layer, in ML-based GMDH, the neurons in each layer were selected and excluded based on the MSE values obtained from analyzing the trained ML models using Eq. 15. Figure 2 displays the ML-based GMDH model with four input variables; two middle layers; and multilayer perceptron (MLP) [33] partial functions of the ML model, with a middle layer containing three neurons.

Various ML models can be used as partial functions in ML-based GMDH. The four conventional models of MLP, SVR [34], random forest (RF) [35], and ELM [36] are considered alternatives to be used as partial functions in this case.

The following hyperparameters in the ML-based GMDH models should be tuned:

1. The selection leverage ( $\alpha$ ) that determines the selection threshold of neurons in each layer.
2. The number of network layers (N-layer).
3. The maximum allowable number of neurons in each layer (Max-Neurons).
4. The type of ML model as the partial function (among MLP, SVR, RF, and ELM).

5. The percentage of observations used for training ( $P_{\text{train}}$ ).

In the next section, the simulation results and comparisons with other ML models are presented.

### Simulation experiment

The performance of the ML-based GMDH model was validated by the five following simulation experiments: a six-dimensional non-polynomial function and four real-world datasets in the UCI repository<sup>1</sup>—namely, household electric power consumption approximation, air-quality approximation, Hungarian chickenpox, and Seoul bike sharing demand. For comparison, the results of the GMDH and ML-based GMDH models are presented separately. The parameters  $\alpha$ , N-layer, Max-Neurons, and  $P_{\text{train}}$  are the same in both the GMDH and ML-based GMDH models and are determined by the commonly used cross-validation method [37]. The results are listed in Table 1.

The evaluation metrics used for comparing the results were the correlation coefficient (R), root mean square error

<sup>1</sup> <https://archive.ics.uci.edu/ml/index.php>.



**Table 1** Similar parameters in the GMDH and ML-based GMDH models

Parameters	$\alpha$	N-layer	Max-Neurons	$P_{\text{train}}$ (versus validation)
Values	0.6	4	15	0.7

(RMSE), mean of absolute errors (MAE), and standard deviation of the absolute errors (STD errors), as follows:

$$\text{MSE}_{\text{train}} = \frac{1}{M} \sqrt{\sum_{i=1}^M (y_i - \hat{y}_i)^2}, \quad (17)$$

$$\text{MAE} = \frac{1}{M} \sum_{i=1}^M |y_i - \hat{y}_i|, \quad (18)$$

$$\text{STD errors} = \sum_{i=1}^M \sqrt{\frac{(y_i - \text{MAE})}{M-1}}, \quad (19)$$

$$R = \frac{\sum_{i=1}^M (y_i - \bar{y}_i) (\hat{y}_i - \bar{\hat{y}}_i)}{\sum_{i=1}^M (y_i - \bar{y}_i)^2 * \sum_{i=1}^M (\hat{y}_i - \bar{\hat{y}}_i)^2}. \quad (20)$$

### Approximation of a six-dimensional non-polynomial function

In this experiment, a six-dimensional non-polynomial function was approximated as follows [38]:

$$f(x_1, x_2, x_3, x_4, x_5, x_6) = x_1 + x_2^{0.5} + x_3 x_4 + 2e^{2(x_5 - x_6)}, \quad (21)$$

where,  $x_1, x_2 \in [1, 5]$ ,  $x_3 \in [0, 4]$ ,  $x_4 \in [0, 0.6]$ ,  $x_5 \in [0, 1]$ , and  $x_6 \in [0, 1.2]$ . Of the total data points, 80% (72,000) were considered for training and the rest (18,000) for testing the network. Figure 3 compares the expected values of function  $f$  with the GMDH and ML-based GMDH outputs (using different types of partial ML). As can be seen in Fig. 3, the different types of ML-based GMDH models presented closer approximations to real values compared with the basic GMDH. For more comparative information, Table 2 provides a detailed comparison between the basic GMDH model and different types of ML-based GMDH models in terms of the evaluation metrics. Table 2 shows that the ML-based GMDH models dominate the basic GMDH in terms of all the evaluation metrics. Overall, the RMSE metric is improved 25%, 18%, 28%, and 27% by MLP-based GMDH, SVR-based GMDH, RF-based GMDH, and ELM-based GMDH, respectively. In terms of the MAE metric, 16%,

10%, 19%, and 18% improvements resulted from the MLP-based GMDH, SVR-based GMDH, RF-based GMDH, and ELM-based GMDH, respectively.

In addition, 42%, 35%, 44%, and 63% improvements are returned by MLP-based GMDH, SVR-based GMDH, RF-based GMDH, and ELM-based GMDH, respectively, in terms of the R metric. Regarding the STD error metric, 38%, 29%, 41%, and 39% improvements are shown by MLP-based GMDH, SVR-based GMDH, RF-based GMDH, and ELM-based GMDH, respectively. It can be concluded that different types of ML-based GMDH models provide better results in approximating the considered six-dimensional non-polynomial function compared with the conventional GMDH.

### Approximation of household electric power consumption

In this task, individual household electric power consumption was approximated. This archive<sup>2</sup> contained 2,075,259 measurements gathered in a house located in Sceaux, France (7 km from Paris), from December 2006 to November 2010 (47 months). Of all the observations, 80% (179,209) were considered for training and the rest (716,835) for testing the network. Table 3 lists the evaluation metrics calculated for the predictions made by the basic GMDH model and the various ML-based GMDH models.

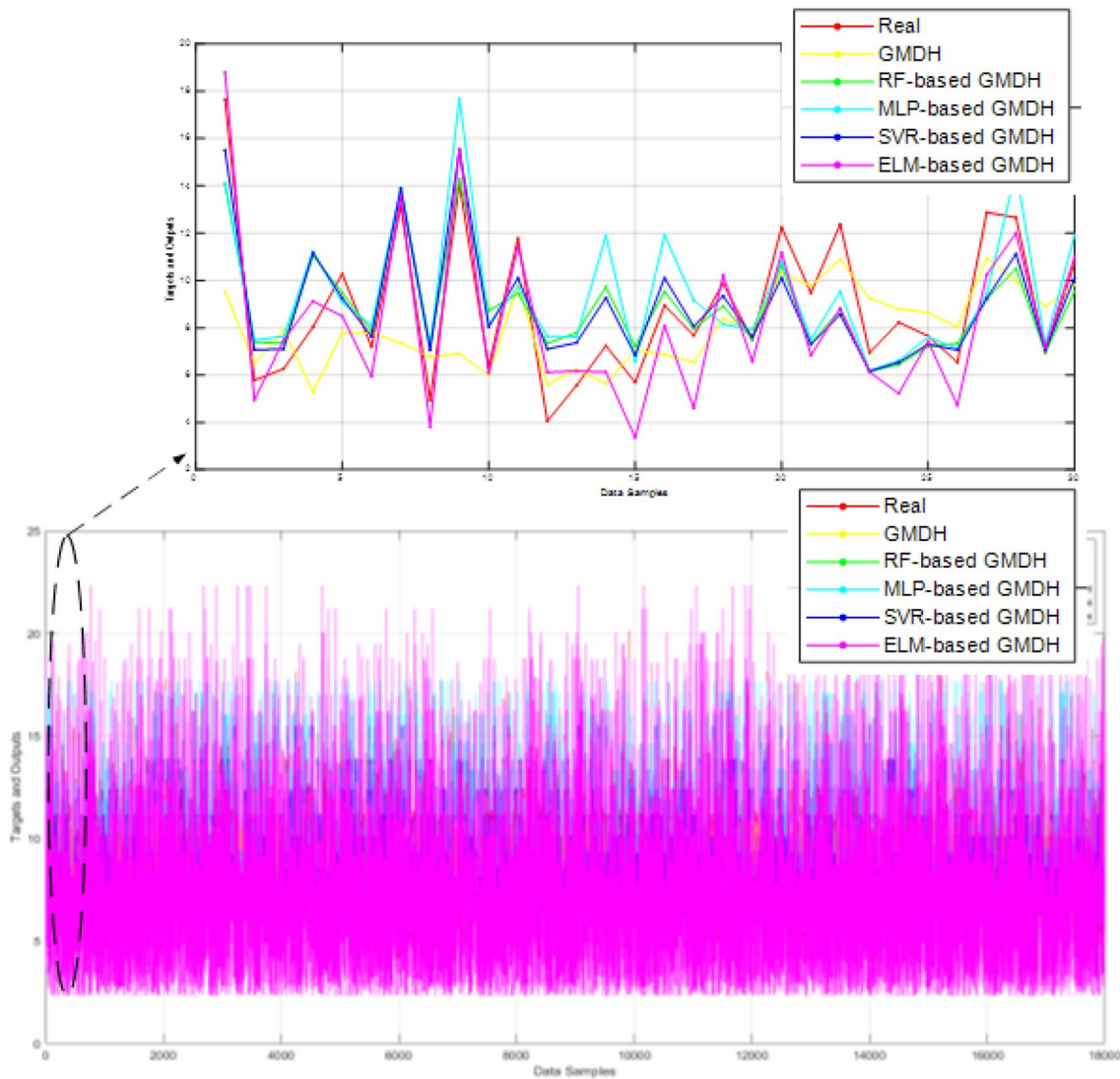
According to the results presented in Table 3, the ML-based GMDH models outperformed the basic GMDH model in approximating the household electric power consumption pattern in terms of all evaluation metrics. In addition, as illustrated in Fig. 4. ELM-based GMDH as an ML-based GMDH performed much better than the basic GMDH method in terms of the RMSE, MAE, and STD error metrics. Even the MLP-based GMDH method with the weakest performance among ML-based GMDH models exhibited good improvement over the base model in terms of all metrics.

### Approximation of air quality

In this task, the PM<sub>2.5</sub> concentration at Shunyi Railway Station in China was approximated. The dataset<sup>3</sup> used for this purpose included the concentrations of pollutants recorded at 12 different railway stations in China from March 1, 2013, to February 28, 2017. Shunyi Station was randomly selected for this evaluation, and its PM<sub>2.5</sub> concentration data were used. Of 34,151 observations (after preprocessing), 80%

<sup>2</sup> <https://archive.ics.uci.edu/ml/datasets/Individual+household+electric+power+consumption>.

<sup>3</sup> <https://archive.ics.uci.edu/ml/datasets/Beijing+Multi-Site+Air-Quality+Data>.



**Fig. 3** Results of the GMDH and ML-based GMDH models on a six-dimensional non-polynomial function

**Table 2** Results of the GMDH and ML-based GMDH models on a six-dimensional non-polynomial function

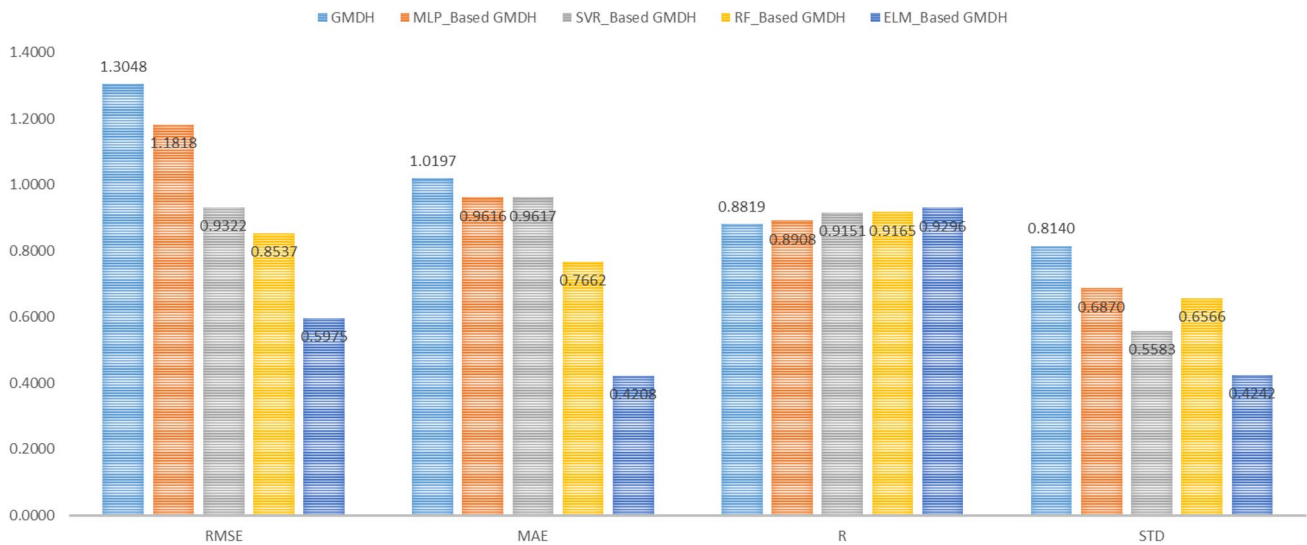
Models	Evaluation metrics			
	RMSE	MAE	R	STD errors
GMDH	2.3281	1.7169	0.5562	1.5725
MLP-based GMDH	1.7259	1.4256	0.7942	0.9730
SVR-based GMDH	1.8991	1.5425	0.7521	1.1078
RF-based GMDH	1.6597	1.3836	0.8059	0.9168
ELM-based GMDH	1.6932	1.4049	0.9103	0.9452

**Table 3** Results of the GMDH and ML-based GMDH models on household electric power consumption

Models	Evaluation metrics			
	RMSE	MAE	R	STD errors
GMDH	251.7513	67.8127	0.0230	242.4639
MLP-based GMDH	24.0462	12.8974	0.9567	20.2962
SVR-based GMDH	21.8306	11.0218	0.9644	18.8454
RF-based GMDH	23.1304	11.9765	0.9500	19.8475
ELM-based GMDH	72.0025	70.6634	0.2048	73.4728

(27,314) were considered for training and 20% (6828) for testing the network. Table 4 shows the calculated evaluation metrics for the predictions made by the basic GMDH model and the various ML-based GMDH models.

According to the results, the ML-based GMDH models outperformed the basic GMDH model in approximating the PM2.5 concentration. As shown in Fig. 5 (for greater clarity, the *R* values are multiplied by 10), all ML-based GMDH



**Fig. 4** Visual comparison of the GMDH and ML-based GMDH models on household electric power consumption

**Table 4** Results of the GMDH and ML-based GMDH models for air quality

Models	Evaluation metrics			
	RMSE	MAE	R	STD errors
GMDH	109.3188	76.6839	0.5669	77.9170
MLP-based GMDH	22.8959	12.9696	0.9603	18.8696
SVR-based GMDH	109.7428	99.3764	0.7025	46.5632
RF-based GMDH	73.6905	14.0540	0.8624	56.2902
ELM-based GMDH	93.3028	65.2028	0.7616	66.7433

performed much better than the basic GMDH method did in terms of  $R$ , RMSE, MAE, and STD error metrics. The extent of the improvement in the ML-based GMDH is clear in approximating the PM<sub>2.5</sub> concentration.

### Approximation of Hungarian chickenpox

In this task, a spatiotemporal dataset of weekly chickenpox cases from Hungary is approximated. This dataset<sup>4</sup> consists of a county-level adjacency matrix and a time series of the county-level reported cases between 2005 and 2015. Of the 520 observations (after preprocessing), 80% (416) were considered for training and 20% (104) for testing the network. Table 5 shows the calculated evaluation metrics for the predictions made by the basic GMDH model and the various ML-based GMDH models.

According to the results, the ML-based GMDH models outperformed the basic GMDH model in approximating chickenpox.

In addition, as observed in Fig. 6 (for greater clarity, the  $R$  values are multiplied by 10), MLP-based GMDH, RF-based GMDH, and ELM-based GMDH exhibited weaker performance compared with the basic GMDH method in terms of RMSE and MAE metrics. Regarding the STD error metric, the basic GMDH method performed better than the other methods did. Only the MLP-based GMDH method presented better results in the RMSE and MAE metrics compared with the basic GMDH method. Overall, in approximating chickenpox, ML-based GMDH did not show considerable improvement in relation to the basic GMDH method. This matter may have arisen either because the complexity in this dataset was negligible/minimal or more observations may have produced a different result.

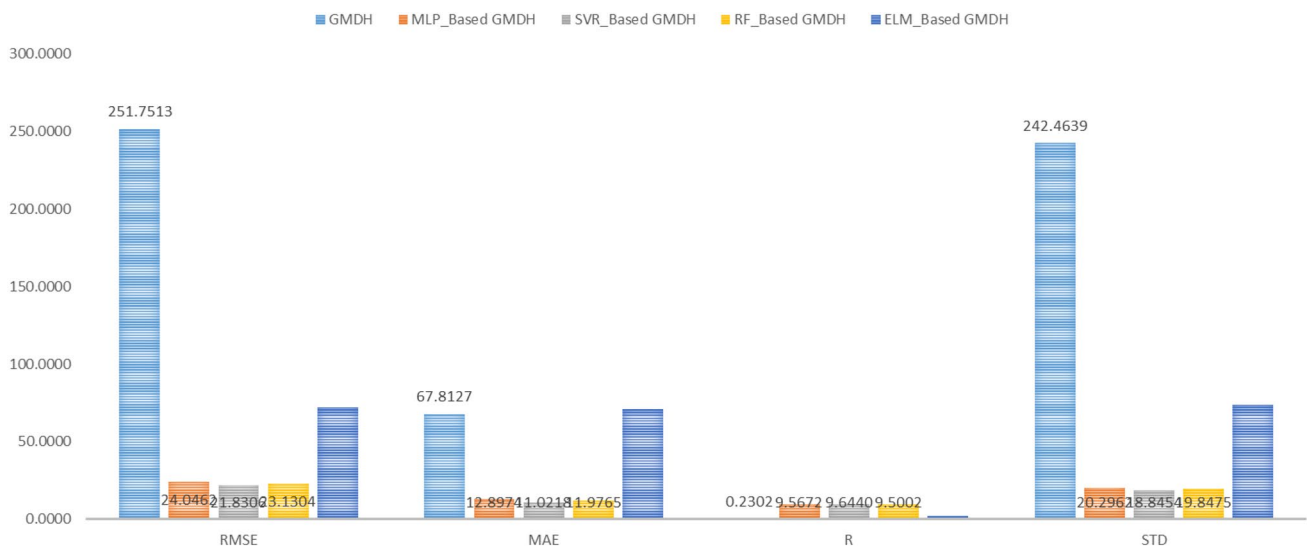
### Approximation of bike-sharing demand

In the next task, the Seoul bike sharing demand was approximated. The dataset<sup>5</sup> contained a count of public bikes rented at each hour in the Seoul bike-sharing system with the corresponding weather data and holiday information. Currently, rental bikes are available in many urban centers for the enhancement of mobility comfort. It is important to make rental bikes available and accessible to the public at the right time because this will lessen users' waiting time. Eventually, providing the city with a stable supply of rental bikes

<sup>4</sup> <https://archive.ics.uci.edu/ml/datasets/Hungarian+Chickenpox+Cases>

<sup>5</sup> <https://archive.ics.uci.edu/ml/datasets/Seoul+Bike+Sharing+Demand>





**Fig. 5** Visual comparison of the GMDH and ML-based GMDH models on approximating the PM2.5 concentration

becomes a major concern. The crucial part is the prediction of the bike count required at each hour for the stable supply of rental bikes. The dataset contains weather information

**Table 5** Results of the GMDH and ML-based GMDH models for approximating chickenpox

Models	Evaluation metrics			
	RMSE	MAE	R	STD errors
GMDH	87.2324	56.2312	0.9310	33.1112
MLP-based GMDH	93.4312	67.3642	0.9411	35.1324
SVR-based GMDH	84.4158	50.9764	0.9220	39.2314
RF-based GMDH	90.1212	63.1030	0.9568	42.1234
ELM-based GMDH	96.5672	69.8091	0.8901	51.1233

(temperature, humidity, wind speed, visibility, dew point, solar radiation, snowfall, rainfall), the number of bikes rented per hour, and date information. Of the 8760 observations (after preprocessing), 80% (7008) were considered for training and 20% (1752) for testing the network. Table 6 shows the calculated evaluation metrics for the predictions made by the basic GMDH model and the various ML-based GMDH models.

According to the results, the ML-based GMDH models outperformed the basic GMDH model in approximating the bike-sharing demand. For comparing results in a graphical shape, evaluation metrics calculated for each method are depicted in Fig. 7 as a bar chart.

As shown in Fig. 7 (for greater clarity, the R values are multiplied by 10), all ML-based GMDH performed much better than the basic GMDH method did in terms of the R, RMSE, MAE, and STD error metrics. The extent of the

improvement made by ML-based GMDH is clear in approximating the bike-sharing demand.

To validate the significant difference between the results obtained by all the methods, we applied a Wilcoxon signed-rank test to four evaluation metrics—RMSE, MAE, R, and STD error. The obtained results are presented in Table 7 in terms of p values.

From the results in Table 7 and the obtained p values, the null hypotheses are rejected and all differences are significant. In other words, all ML-based GMDH models showed significantly better results in approximating the power consumption, PM2.5 concentration, and bike-sharing demand compared with the conventional GMDH model. Only in approximating Hungarian chickenpox did the ML-based GMDH models fail to show better results in all evaluation metrics. Regarding the number of observations in this dataset, the performance of ML-based GMDH models—with their complexities related to the basic GMDH model—may have been affected by underfitting. This is suggested because the ML-based GMDH models showed significantly better results in three cases with a greater number of observations.

## Conclusion

The approximation capability of the GMDH model was improved by combining it with conventional ML models—namely, SVR, RF, MLP, and ELM. To this end, the basic partial functions (polynomial) in the GMDH model, which are used as transfer functions in neurons, were replaced by ML models. Given the GMDH mechanism and the role of the polynomial partial functions in this method, the ML models were considered black boxes in the sequential

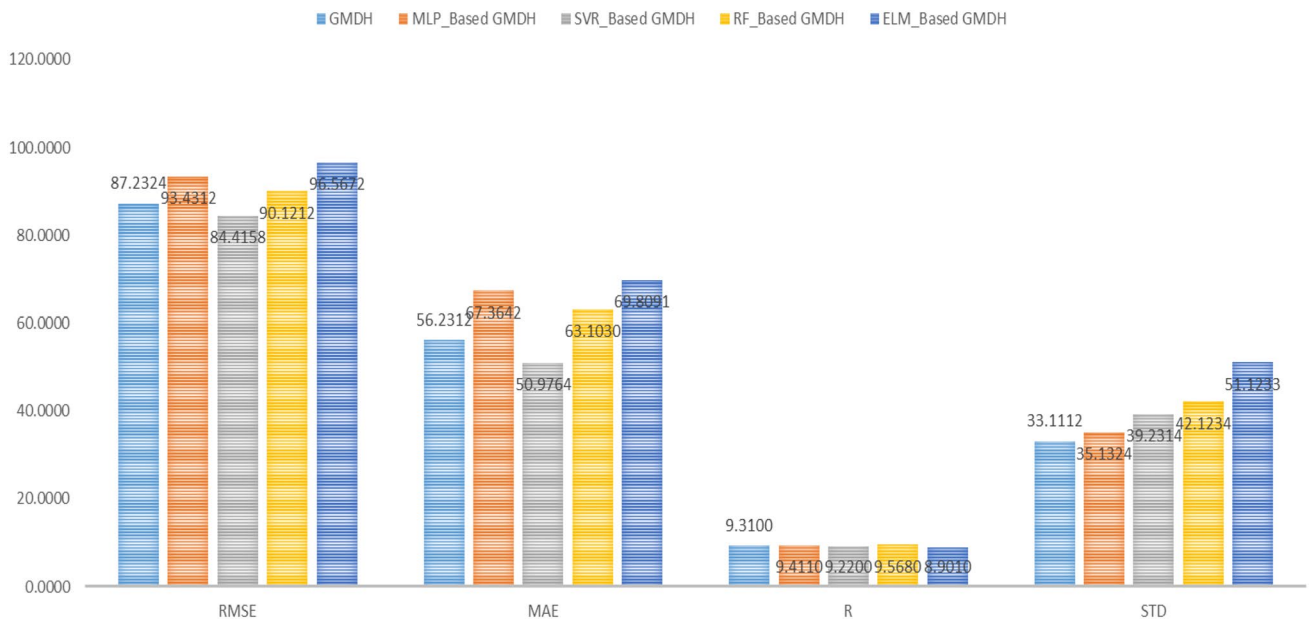


Fig. 6 Visual comparison of the GMDH and ML-based GMDH models on approximating chickenpox

Table 6 Results of the GMDH and ML-based GMDH models bike-sharing demand

Models	Evaluation metrics			
	RMSE	MAE	R	STD Errors
GMDH	53.1576	42.1134	0.7852	17.5543
MLP-based GMDH	24.4465	34.8752	0.8912	12.2314
SVR-based GMDH	21.3356	29.1735	0.9014	9.5327
RF-based GMDH	27.9872	32.5140	0.9143	11.1432
ELM-based GMDH	31.1097	34.8202	0.9324	13.1432

structure of the GMDH to approximate the relationship between the input and output pairs. The simulation results on a non-polynomial function and four real-world datasets confirmed the better performance of the ML-based GMDH models compared with the GMDH model in terms of the RMSE, MAE, R, and STD error. In the proposed ML-based GMDH model, in each neuron, an ML model is trained between two (or more) inputs and targets. This mechanism may be time-consuming in cases with many inputs and observations. This matter can be addressed by incorporating information theory-based methods and feature selection approaches into the models in future work.

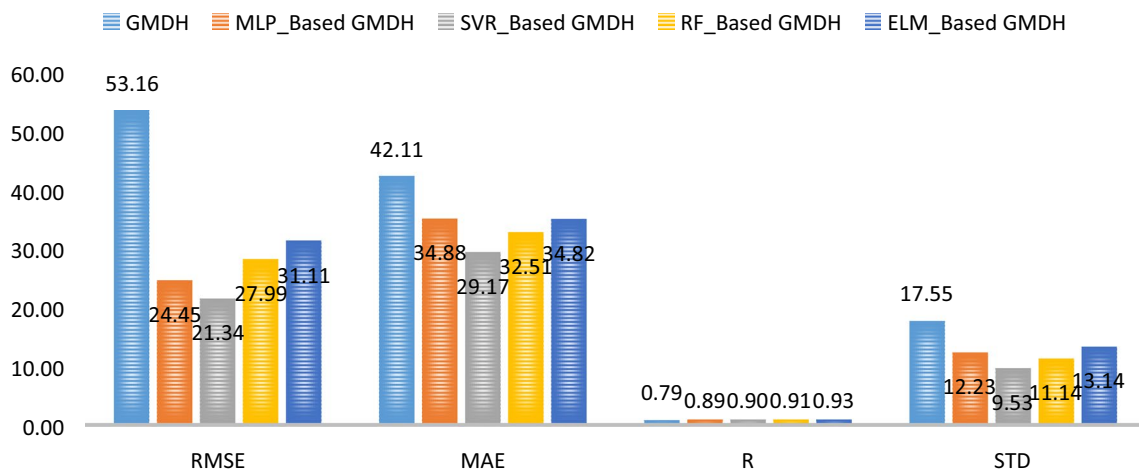


Fig. 7 Visual comparison of the GMDH and ML-based GMDH models on approximating bike sharing demand

**Table 7** Wilcoxon signed-rank test for evaluating significance on approximating results

	Evaluation metrics	MLP-based GMDH	SVR-based GMDH	RF-based GMDH	ELM-based GMDH
GMDH	RMSE	0.375	0.125	0.375	1
	MAE	0.625	0.125	0.75	0.875
	R	0.125	0.25	0.125	0.25
	STD Errors	0.375	0.375	0.625	0.625

## Declarations

**Conflict of interest** On behalf of all authors, the corresponding author states that there is no conflict of interest.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Ivakhnenko AG (1968) The group method of data of handling; a rival of the method of stochastic approximation. *Soviet Auto Contr* 13(1):43–55
- Mehra RK (1977) Group method of data handling (GMDH): review and experience. In: 1977 IEEE Conference on Decision and Control including the 16th Symposium on Adaptive Processes and A Special Symposium on Fuzzy Set Theory and Applications vol 5 No 1, pp 29–34
- Anastasakis L, Mort N (2001) The development of self-organization techniques in modelling: a review of the group method of data handling (GMDH). In: Research report-university of sheffield department of automatic control and systems engineering
- Ohtani T, Ichihashi H, Miyoshi T, Nagasaka K (1998) Structural learning with M-apoptosis in neurofuzzy GMDH. In: 1998 IEEE International Conference on Fuzzy Systems Proceedings. IEEE World Congress on Computational Intelligence (Cat. No. 98CH36228). vol 6. No 2, pp 1265–1270
- Kondo T (2006) Revised gmdh-type neural network algorithm with a feedback loop identifying sigmoid function neural network. In: Proceedings of the ISCIE International Symposium on Stochastic Systems Theory and its Applications. vol 6, pp 137–142
- Elattar EE, Goulermas JY, Wu QH (2011) Generalized locally weighted GMDH for short-term load forecasting. *IEEE Trans Syst Man Cybern Part C (Appl Rev)* 42(3):345–356
- Zhang M, He C, Liatsis P (2012) A D-GMDH model for time series forecasting. *Expert Syst Appl* 39(5):5711–5716
- Shahsavari A, Khanmohammadi S, Karimipour A, Goodarzi M (2019) A novel comprehensive experimental study concerned synthesizes and prepare liquid paraffin-Fe<sub>3</sub>O<sub>4</sub> mixture to develop models for both thermal conductivity & viscosity: a new approach of GMDH type of neural network. *Int J Heat Mass Transf* 131:432–441
- Band SS, Mohammadzadeh A, Csiba P, Mosavi A, Varkonyi-Koczy AR (2020) Voltage regulation for photovoltaics-battery-fuel systems using adaptive group method of data handling neural networks (GMDH-NN). *IEEE Access* 8:213748–213757
- Mahdavi-Meymand A, Zounemat-Kermani M (2020) A new integrated model of the group method of data handling and the firefly algorithm (GMDH-FA): application to aeration modelling on spillways. *Artif Intell Rev* 53(4):2549–2569
- Najafzadeh M, Saberi-Movahed F, Sarkamaryan S (2018) NF-GMDH-based self-organized systems to predict bridge pier scour depth under debris flow effects. *Mar Georesour Geotechnol* 36(5):589–602
- Mo L, Xie L, Jiang X, Teng G, Xu L, Xiao J (2018) GMDH-based hybrid model for container throughput forecasting: Selective combination forecasting in nonlinear subseries. *Appl Soft Comput* 62:478–490
- Najafzadeh M, Saberi-Movahed F (2019) GMDH-GEP to predict free span expansion rates below pipelines under waves. *Mar Georesour Geotechnol* 37(3):375–392
- Harandizadeh H, Armaghani DJ, Khari M (2019) A new development of ANFIS–GMDH optimized by PSO to predict pile bearing capacity based on experimental datasets. *Eng Comput* 5(37):685–700
- Armaghani DJ, Hasanipanah M, Amnieh HB, Bui DT, Mehrabi P, Khorami M (2019) Development of a novel hybrid intelligent model for solving engineering problems using GS-GMDH algorithm. *Eng Comput* 13(36):1379–1391
- Harandizadeh H, Armaghani DJ, Mohamad ET (2020) Development of fuzzy-GMDH model optimized by GSA to predict rock tensile strength based on experimental datasets. *Neural Comput Appl* 32(17):14047–14067
- Xu L, Wang X, Bai L, Xiao J, Liu Q, Chen E, Luo B (2020) Probabilistic SVM classifier ensemble selection based on GMDH-type neural network. *Pattern Recogn* 106:107373
- Jeddi S, Sharifian S (2020) A hybrid wavelet decomposer and GMDH-ELM ensemble model for Network function virtualization workload forecasting in cloud computing. *Appl Soft Comput* 88:105940
- Elbaz K, Shen SL, Zhou A, Yin ZY, Lyu HM (2021) Prediction of disc cutter life during shield tunneling with AI via the incorporation of a genetic algorithm into a GMDH-type neural network. *Engineering* 7(2):238–251
- Sharif MI, Khan MA, Alhussain M, Aurangzeb K, Raza M (2021) A decision support system for multimodal brain tumor classification using deep learning. *Complex Intell Syst*:1–14.
- Ketu S, Mishra PK (2021) Scalable kernel-based SVM classification algorithm on imbalance air quality data for proficient healthcare. *Complex Intell Syst*:1–19.
- Shankar K, Perumal E (2021) A novel hand-crafted with deep learning features based fusion model for COVID-19 diagnosis

- and classification using chest X-ray images. *Complex Intell Syst* 7(3):1277–1293
23. Hesarian MS, Eshkevari M, Jahangoshai Rezaee M (2020) Angle analysis of fabric wrinkle by projected profile light line method, image processing and neuro-fuzzy system. *Int J Comput Integr Manuf* 33(10–11):1167–1184
  24. Sabri-Laghaie K, Sharifpour A, Eshkevari M, Aghbolaghi M (2021) Early detection of product reliability based on the parameters of the production line and warranty data. *Int J Reliabil Qual Safety Eng* 13(1):2150035–2150047
  25. Eshkevari M, Rezaee MJ, Zarinbal M, Izadbakhsh H (2021) Automatic dimensional defect detection for glass vials based on machine vision: a heuristic segmentation method. *J Manuf Process* 68:973–989
  26. Sabri-Laghaie K, Eshkevari M, Fathi M, Zio E (2019) Redundancy allocation problem in a bridge system with dependent subsystems. *Proc Inst Mech Eng Part O J Risk Reliab* 233(4):658–669
  27. Lu J, Liu A, Song Y, Zhang G (2020) Data-driven decision support under concept drift in streamed big data. *Complex Intell Syst* 6(1):157–163
  28. Naderpour H, Mirrashid M (2020) Moment capacity estimation of spirally reinforced concrete columns using ANFIS. *Complex Intell Syst* 6(1):97–107
  29. Onari MA, Yousefi S, Rabieepour M, Alizadeh A, Rezaee MJ (2021) A medical decision support system for predicting the severity level of COVID-19. *Complex Intell Syst*:1–15
  30. Yi S, Liu X (2020) Machine learning based customer sentiment analysis for recommending shoppers, shops based on customers' review. *Complex Intell Syst* 6(3):621–634
  31. Assaleh K, Shanableh T, Kheil YA (2013) Group method of data handling for modeling magnetorheological dampers 4(1):27845–27854
  32. Band SS, Al-Shourbaji I, Karami H, Karimi S, Esfandiari J, Mosavi A (2020) Combination of group method of data handling (GMDH) and computational fluid dynamics (CFD) for prediction of velocity in channel intake. *Appl Sci* 10(21):7521
  33. Taud H, Mas JF (2018) Multilayer perceptron (MLP). *Geomatic approaches for modeling land change scenarios*. Springer, Cham, pp 451–455
  34. Awad M, Khanna R (2015) Support vector regression. *Efficient learning machines*. Apress, Berkeley, pp 67–80
  35. Hong WH, Yap JH, Selvachandran G, Thong PH (2021) Forecasting mortality rates using hybrid Lee-Carter model, artificial neural network and random forest. *Complex & Intelligent Systems* 7(1):163–189
  36. Huang GB, Zhu QY, Siew CK (2006) Extreme learning machine: theory and applications. *Neurocomputing* 70(1–3):489–501
  37. Zhang L, Zhou W, Jiao L (2004) Wavelet support vector machine. *IEEE Trans Syst Man Cybern Part B (Cybern)* 34(1):34–39
  38. Zhang Q, Benveniste A (1992) Wavelet networks. *IEEE Trans Neural Netw* 3(6):889–898

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