Modeling Earth Systems and Environment
https://doi.org/10.1007/s40808-022-01573-6

ORIGINAL ARTICLE

Modeling air pollution by integrating ANFIS and metaheuristic algorithms

Aynur Yonar[1] · **Harun Yonar**[2]

Received: 18 August 2022 / Accepted: 13 October 2022
© The Author(s), under exclusive licence to Springer Nature Switzerland AG 2022

Abstract
Air pollution is increasing for many reasons, such as the crowding of cities, the failure of planning to consider the benefit of society and nature, and the non-implementation of environmental legislation. In the recent era, the impacts of air pollution on human health and the ecosystem have become a primary global concern. Thus, the prediction of air pollution is a crucial issue. ANFIS is an artificial intelligence technique consisting of artificial neural networks and fuzzy inference systems, and it is widely used in estimating studies. To obtain effective results with ANFIS, the training process, which includes optimizing its premise and consequent parameters, is very important. In this study, ANFIS training has been performed using three popular metaheuristic methods: Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and Differential Evolution (DE) for modeling air pollution. Various air pollution parameters which are particular matters: $PM_{2.5}$ and PM_{10}, sulfur dioxide (SO_2), ozone (O_3), nitrogen dioxide (NO_2), carbon monoxide (CO), and several meteorological parameters such as wind speed, wind gust, temperature, pressure, and humidity were utilized. Daily air pollution predictions in Istanbul were obtained using these particular matters and parameters via trained ANFIS approaches with metaheuristics. The prediction results from GA, PSO, and DE-trained ANFIS were compared with classical ANFIS results. In conclusion, it can be said that the trained ANFIS approaches are more successful than classical ANFIS for modeling and predicting air pollution.

Keywords Air pollution · ANFIS · Artificial intelligence · Metaheuristics

Introduction

Increasing environmental problems threaten nature and human health. Air pollution is at the forefront of this threat. Air, the primary source of life, is indispensable for humans and living things. Therefore, air pollution is a global threat that significantly impacts human health and ecosystems. Air pollution can occur from natural causes such as forest fires, earthquakes, volcanic activities, swamps, and human activities such as industrialization, heating, transportation, and energy production. In addition, population growth, increasing urbanization, industrialization, drought, topographic conditions, inversion, and climatic features affect air pollution.

According to the World Health Organization (WHO), approximately 7 million deaths are caused by outdoor and indoor air pollution each year. Air pollution is a significant environmental risk to health. Reducing air pollution reduce premature deaths and diseases like stroke, heart diseases, lung cancer, chronic and acute respiratory diseases, and asthma. These premature deaths and diseases result from exposure to 2.5 microns or less of particulate matter ($PM_{2.5}$), one of the most harmful components of air pollution (Afghan et al. 2022; Barthwal and Acharya 2022).

The COVID-19 virus pandemic, which entered our lives in the last months of 2019, also revealed the importance of the relationship between public health and the environment. Studies indicate that people exposed to long-term air pollution have a higher risk of contracting and adversely affecting viruses such as COVID-19 due to emerging chronic diseases. After all these developments, air quality management is becoming an increasingly important issue for citizens and decision-makers worldwide.

✉ Harun Yonar
hyonar@selcuk.edu.tr

Aynur Yonar
aynursahin@selcuk.edu.tr

[1] Faculty of Science, Department of Statistics, Selçuk University, Konya, Turkey

[2] Faculty of Veterinary Medicine, Department of Biostatistics, Selçuk University, Konya, Turkey

Published online: 28 October 2022

Adaptive Network Fuzzy Inference System (ANFIS) is one of the most popular neuro-fuzzy systems. It is a hybrid artificial intelligence technique which is the combination of fuzzy logic and artificial neural networks (ANN). ANFIS does not explain as to the structure of the physical process of the data analyzed during the modeling phase, but it is capable of extracting relationship between input and output of process. Hence, it has been widely used in solving a range of air pollution prediction problems (Yılmaz et al. 2022).

Many studies have been carried out related to the prediction of air pollution in the literature, and various statistical and computational methods have been used for modeling air pollution. Polat and Durduran (2012) present the combination of a data preprocessing called output-dependent data scaling (ODDS) and ANFIS for predicting the PM_{10} concentration values for the city of Konya in Turkey. Kemal Polat (2012) proposed a novel feature scaling method called neighbor-based feature scaling (NBFS) and combined it with ANN and ANFIS for predicting the SO_2 concentration value for Konya province in Turkey. Mirzaei et al. (2019) utilized a linear model (LR), ANN, and hybrid models, including general regression neural network (GRNN) and ANFIS, to investigate the spatial and temporal variability relationship between $PM_{2.5}$ concentrations for Tehran. Ghasemi and Amanollahi (2019) developed ANFIS with the collinearity tests and forward selection (FS) technique to reduce the computational cost and time for predicting five daily air pollutants: PM_{10}, SO_2, NO_2, CO, and O_3 in the atmosphere of Kermanshah city. Cekim (2020) forecasted the values of PM_{10} for the most polluted cities in Turkey using some time series models, which are the autoregressive integrated moving average (ARIMA), error, trend, and seasonal (ETS), and singular spectrum analysis (SSA). Ordóñez-De León et al. (2020) proposed combining a neuro-fuzzy-based method with particle swarm optimization to predict the airborne particulate matter (PM_{10}). Purnomo and Anugerah (2020) predicted six air pollutants, including SO_2, NO_2, O_3, HC, Pb, and PM_{10}, in Yogyakarta using an adaptive neuro-fuzzy inference system (ANFIS). Zhou et al. (2020) configured BPNN and ANFIS to establish deterministic forecast models for the regional $PM_{2.5}$ concentrations of Taipei City in Taiwan. Amanollahi and Ausati (2020a, b) predicted PM_{10} concentration in the air of Tehran by various models, which are multiple linear regression, MLR, two hybrid models: ANFIS and empirical mode decomposition, and general regression neural network (EEMD-GRNN), and multi-layer perceptron (MLP) mode. Amanollahi and Ausati (2020a, b) used a variety of models for predicting $PM_{2.5}$ concentrations, including multiple linear regression, multi-layer perceptron (nonlinear model), and an ensemble empirical mode decomposition and general regression neural network (EEMD-GRNN) and ANFIS. Bhardwaj and Pruthi (2020) used the ANFIS by combining PSO and ANFIS to perform predictive analysis of air pollutant—$PM_{2.5}$ for Shadipur, Delhi. Furthermore, they decomposed the non-stationary $PM_{2.5}$ time series via wavelet transform. Tunckaya (2020) made a performance analysis of a novel air pollution forecasting system design in a Turkish cement plant via ANFIS, ANN, and MLR methods. Shukura (2020) combined the NF as a nonlinear intelligent method with MLR in a hybrid MLR-NF method to improve PM_{10} forecasts for Malaysians. Tauqir and Kashif (2022) examined the impact of COVID-19 restrictions on the air quality of Lahore city of Pakistan with asymmetrical Granger causality tests. Their conclusion is to control unnecessary production and consumption activities to reduce air pollution in the city.

In this paper, we have used the ANFIS, one of the most popular artificial intelligence techniques, for air pollutant—$PM_{2.5}$ prediction. Furthermore, various metaheuristic methods, such as GA, PSO, and DE have been utilized in training ANFIS to improve the performance of ANFIS. Air pollutant—$PM_{2.5}$ values belonging to İstanbul province have been predicted by trained ANFIS structures. To evaluate the performance of suggested trained ANFIS methods, mean square error (MSE), root mean square error (RMSE), and mean absolute percentage error (MAPE) values have been used.

The rest of the paper is organized as follows. In Sect. 2, ANFIS and some metaheuristic methods: GA, PSO, and DE, used in the training of ANFIS in this study are introduced. In Sect. 3, prediction results of air pollution—$PM_{2.5}$ obtained via classical ANFIS and trained ANFIS approaches by GA, PSO, and DE are presented and compared. Finally, conclusions are discussed in Sect. 5.

Data sources and method

This study was carried out to model air pollutant $PM_{2.5}$, the most important indicator of air pollution in Istanbul. Data consist of the daily meteorological data: sulfur dioxide (SO2), ozone (O3), nitrogen dioxide (NO2), carbon monoxide (CO), and several meteorological parameters: wind speed, wind gust, temperature, pressure, and humidity. The air quality data set was obtained from Air Quality Open Data Platform ("Air Quality Open Data Platform" 2021). $PM_{2.5}$ values were estimated for the province of Istanbul using ANFIS constructs trained with GA, PSO, and DE.

Adaptive network fuzzy inference system

Adaptive network fuzzy inference system (ANFIS), developed by Jang (1993), is one of the most popular neuro-fuzzy systems, which is the primary technique of artificial intelligence. It consists of an artificial neural network and fuzzy inference system. It combines the advantages of these two methods by enabling the artificial neural network to take its decision-making mechanisms from fuzzy logic and the learning capabilities of fuzzy logic from the artificial neural network.

Different types of ANFIS are introduced regarding fuzzy inference systems such as Mamdani, Sugeno, and Tsukamoto. ANFIS structure-based Sugeno fuzzy model, which is the most widely used in the literature, has been utilized in this study (Stanley et al. 2015).

The network structure of ANFIS consists of two parts called premise and consequent parts. The parameters belonging to these parts are used in ANFIS training. These parameters are determined through an optimization algorithm. The existing input–output data couples and if-then fuzzy rules are utilized during the ANFIS training. The difference between the output obtained during training and the system's actual output gives the error. To minimize errors, ANFIS parameters are continuously updated, and thus the most optimum structure is created. An ANFIS structure with two inputs, one output, four membership functions, and four rules are given in Fig. 1 (Bhagowati et al. 2022).

ANFIS structure consists of five layers. The layer structure of ANFIS given in Fig. 1 is explained below (Jang 1993; Stanley et al. 2015):

Layer 1: This layer, called as fuzzification layer, uses membership functions to obtain fuzzy clusters from the values of inputs. With this transaction, membership values in [0,1] are calculated. Different membership functions such as generalized bell function, triangle, trapezium, Gaussian, sigmoid, etc. may be used to find membership values. To set the form of the membership function, parameters like $\{a_i, b_i, c_i\}$ are used. These are called premise or antecedent parameters and are utilized in ANFIS training. The membership degrees of each membership function are calculated as follows:

$$\mu_{A_i}(x) = gbellmf(x;a,b,c) = \frac{1}{1 + \left|\frac{x-c}{a}\right|^{2b}}, \tag{1}$$

$$O_i^1 = \mu_{A_i}(x). \tag{2}$$

Layer 2: This layer, called the rule layer, finds firing strengths (w_i) for each rule using the membership values obtained in the fuzzification layer. w_i values are computed by multiplying the membership values as follows:

$$O_i^2 = w_i = \mu_{A_j}(x).\mu_{B_j}(y),\ j = 1,2; i = 1,2,3,4. \tag{3}$$

Layer 3: This layer, called as normalization layer, calculates the normalized firing strengths ($\overline{w}_i$) for each rule using the firing stress found in the previous layer. Normalized firing strength of the ith rule is the ratio of the firing strength of ith rule to the total firing strengths and is calculated as follows:

$$O_i^3 = \overline{w}_i = \frac{w_i}{\sum_{i=1}^{4} w_i},\ i = 1,2,3,4. \tag{4}$$

Layer 4: This layer, called as defuzzification layer, computes the output of each rule by multiplication of the normalized firing strengths and a first-order polynomial. Calculation of the outputs is given in (4).

$$O_i^4 = \overline{w}_i f_i = \overline{w}_i\left(p_i x + q_i y + r_i\right) \tag{5}$$

Here, $\{p_i, q_i, r_i\}$ are the parameter set in the first-order polynomial. These parameters used in ANFIS training are called conclusion or consequent parameters.

Layer 5: This layer, called as summation layer, obtains the actual output of ANFIS by summing the outputs obtained in the defuzzification layer.

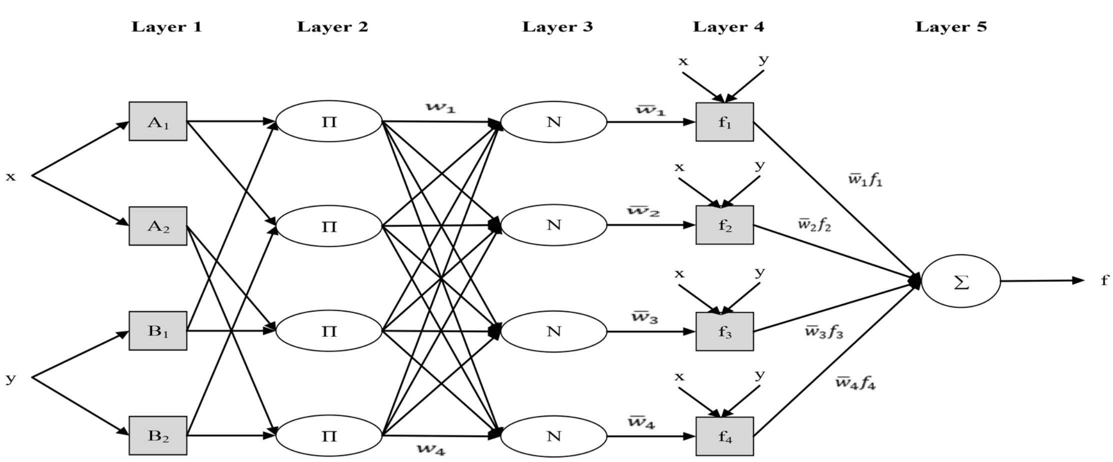

Fig. 1 The ANFIS structure that has two inputs and one output

$$O_i^5 = \sum_{\bar{i}} \overline{w_i} f_i = \frac{\sum_{\bar{i}} w_i f_i}{\sum_{\bar{i}} w_{ii}} \quad (6)$$

The difference between the actual output and the predicted output of ANFIS is the error. The error value is low in the successful ANFIS model. However, it can be seen as a disadvantage that the weight values of the ANFIS model cannot be explained, that is, a clear model cannot be written. Despite this, it is widely used in the literature because it has many advantages such as learning using examples, requires no assumption on the underlying model, working with insufficient and incomplete information, and being aware of machine learning (Karaboga and Kaya 2019; Abbaspour-Gilandeh and Abbaspour-Gilandeh 2019; Pahlavani et al. 2017).

Metaheuristic algorithms

Metaheuristic methods allow tackling large-size problem instances by delivering satisfactory solutions in a reasonable time. These generally start by generating a random initial solution or population and then loop over an iteration process to make the solution or population evolves. For D-dimensional optimization problem, $\vec{x}_i^k = [x_{1,i}^k, x_{2,i}^k, ..., x_{D,i}^k]$ indicates the i th vector of the population at iteration k. The initial population for each element of the vector i is generated as follows through the prescribed lower limit ($x_{j,\min}$) and upper limit ($x_{j,\max}$), which are known as search space, (Talbi 2009; Yang 2010).

$$x_{j,i}^0 = x_{j,\min} + rand_{ij}[0,1]\left\{x_{j,\max} - x_{j,\min}\right\} \quad (7)$$

Here, $rand_{ij}[0,1]$ is a uniformly distributed random variable in the range $[0,1]$(Price et al. 2006; Talbi 2009).

The three metaheuristic algorithms GA, PSO, and DE used in training the ANFIS are shortly explained in the following subsections.

Genetic algorithm (GA)

Genetic algorithm (GA) developed by Holland (1975) is a valuable and efficient search method to obtain approximate solutions for optimization problems (Goldberg and Holland 1988; Talbi 2009; Yang 2014).

The GA starts by generating a random initial population, and then it loops over an iteration process for evolving the population. Each iteration comprises selection, reproduction involving the crossover and mutation operators, evolution, and replacement stages (Talbi 2009; Yalçınkaya et al. 2018; Yang 2010).

Steps of the GA

Step 1. Defining the fitness function, the search space, and the GA parameters: population size (NP), mutation probability (p_m), crossover probability (p_c), and mutation rate (m).

Step 2. Generating the random initial population $\vec{x}_i^0 = [x_{1,i}^0, x_{2,i}^0, ..., x_{D,i}^0]$ via the predetermined search space and calculating fitness function value for the initial population.

Step 3. Selecting a pair of parent solutions from the current population, generating two offspring through the crossover operator, and evaluating the fitness function value of these individuals.

Step 4. Selecting a parent and generating new candidate solution/individual using the mutation operator with the mutation probability (p_m) and evaluating the fitness function value of these individuals.

Step 5. Creating a new population by combining all solutions and applying truncation to select the best individuals as the population size and replacing the new population with the senior population for the next generation.

Step 6. Evolving the population until the stopping criterion is satisfied. The solution with the best fitness function value at the last iteration is the best solution.

Particle swarm optimization (PSO)

Particle swarm optimization (PSO) introduced by Eberhart and Kennedy (1995) is a biologically inspired technique derived from the collective behavior of bird flocking and fish schooling. The population composes of a set of particles. Each particle records its own personal best position (pbest), and knows the best positions found by all particles in the swarm (gbest). Then, all particles update their velocity and position in each iteration.

The velocity and the new position of each particle at iteration $k+1$ can be calculated as follows, respectively:

$$v_i^{k+1} = wv_i^k + c_1 r_1 \left(pbest_i^k - x_i^k\right) + c_2 r_2 \left(gbest^k - x_i^k\right), \quad (8)$$

$$x_i^{k+1} = x_i^k + v_i^k. \quad (9)$$

In Eqs. (8) and (9), v_i^k is the velocity of individual i at iteration k, w is the inertia weight, c_1 and c_2 are the acceleration coefficients, r_1 and r_2 are random numbers uniformly distributed between 0 and 1, x_i^k is the position of individual i at iteration k, $pbest_i^k$, is the best position of individual i until iteration k, $gbest^k$ is the best position of the group until iteration g(Talbi 2009; Yang 2010).

Steps of the PSO algorithm

Step 1. Defining the fitness function, the search space, and PSO parameters: inertia weight (w) and acceleration coefficients c_1 and c_2, and particle (population) size.

Step 2. Generating the random initial population $\vec{x}_i^0 = [x_{1,i}^0, x_{2,i}^0, ..., x_{D,i}^0]$ via the predetermined search space and calculating the fitness function value of each solution of the population.

Step 3. Recording personal best position (pbest) for each particle and finding the best positions by all particles in the swarm (gbest).

Step 4. Calculating the particle velocity according to Equation (7) and updating the particle position with Equation (8) for each population solution.

Step 5. Replacing the current population with the new population. If the stopping criterion is not satisfied, go to step 3, else the solution with the best fitness function value at the last iteration is the best solution.

Differential evolution (DE)

Differential evolution (DE), developed by Storn (1996) and Storn and Price (1997) is one of the most successful approaches for continuous optimization problems. Similar to GA, DE algorithm uses crossover, mutation, and selection operators. However, DE uses the mutation operator to obtain better solutions, while GA uses the crossover operator (Gui et al. 2019; Price et al. 2006). In this study, the standard variant of the DE, $DE/rand/1/bin$, has been used (Das et al. 2016).

Steps of the DE algorithm

Step 1. Defining the objective function, the search space and the DE parameters: population size ($NP \geq 4$), scaling factor ($F \in (0, 1]$), and crossover factor (C_r).

Step 2. Generating the random initial population $\vec{x}_i^0 = [x_{1,i}^0, x_{2,i}^0, ..., x_{D,i}^0]$ via the predetermined search space and calculating the fitness function value of each population solution.

Step 3. Choosing randomly three distinct vectors $c_1, c_2, c_3; c_1 \neq c_2 \neq c_3 \neq i \in (0, NP)$ for each solution of the population and generating new donor vector ($\vec{v}_i^k$) by mutation scheme given by

$$\vec{v}_i^k = \vec{x}_{c1}^k + F\left(\vec{x}_{c2}^k - \vec{x}_{c3}^k\right). \tag{10}$$

Step 4. Generating a random index $j_{rand} \in [0, D]$ and applying the crossover operator given by Eq. (11) to increase the population's diversity.

$$\vec{u}_{j,i}^{k+1} = \begin{cases} \vec{v}_{j,i}^k, & if\ rand_{i,j}(0,1) \leq CR\ \ or\ \ j = j_{rand} \\ \vec{x}_{j,i}^k, & otherwise \end{cases} \tag{11}$$

Step 5. Appling the selection scheme given by Equation (12) to determine the solutions to be transferred to the next generation.

$$\vec{x}_i^{k+1} = \begin{cases} \vec{u}_i^k, & f(\vec{u}_i^k) \geq f(\vec{x}_i^k) \\ \vec{x}_i^k, & f(\vec{u}_i^k) < f(\vec{x}_i^k) \end{cases} \tag{12}$$

Step 6. Replacing the current population with the new population. If the stopping criterion is not satisfied, go to Step 3. Else the solution with the best fitness function value at the last iteration is the best solution.

Training ANFIS using the metaheuristic methods

ANFIS training means determining its premise and consequence parameters using an optimization algorithm. Premise parameters $\{a_i, b_i, c_i\}$ belong to membership functions on the first layer. Consequent parameters $\{p_i, q_i, r_i\}$ also belong to the first-order polynomial fourth layer. Successful training is essential to achieve effective results with ANFIS. In the first developed classical ANFIS method, a hybrid learning approach was used for training. This learning approach determined premise parameters by gradient descent (GD) algorithm while consequence parameters were determined by the least square estimation (LSE) method. However, there is a risk of getting stuck at the local minimum since these methods are derivative-based. So, using metaheuristic methods instead of derivative-based algorithms provides more efficient performance. Due to such reasons, it is recommended to use some of the metaheuristic algorithms such as GA, PSO, and DE for the training of ANFIS in this study. The best model is created by optimizing the ANFIS parameters with the GA, PSO, and DE algorithms to obtain the lowest differences between the actual output values and the predicted output values derived from ANFIS.

Results

Predicting of air pollutant $PM_{2.5}$ value, which is one of the most crucial indicators of air pollution, is a vital process in environmental research. To predict air pollutant $PM_{2.5}$

values in the air of İstanbul province, various trained-ANFIS structures by GA, PSO, and DE have been used. The target/output variable in these structures is the values of $PM_{2.5}$, and predictor/input variables are the daily meteorological data which consist of sulfur dioxide (SO_2), ozone (O_3), nitrogen dioxide (NO_2), carbon monoxide (CO), and several meteorological parameters such as wind speed, wind gust, temperature, pressure, and humidity, for the year 2019 in İstanbul.

Table 1 presents the results of descriptive statistics for the air pollutants. The minimum value for $PM_{2.5}$ considered as target variable in this study was recorded 25 ug/m^3 and the maximum value was 157.00 ug/m^3, the mean value 62.83 ug/m^3 and standard deviation value is 22,21,165 ug/m^3 in İstanbul for the year 2019. As can be seen from these results, it has skewed values. Therefore, using a machine learning technique like ANFIS to forecast the $PM_{2.5}$ would be advantageous. Descriptive statistics of other air pollutants can be similarly read from this table.

Before ANFIS models were created, the normalization method was used to clear irrelevantly or too far from traditional value and thus increase the accuracy of results and achieve faster convergence. The data were normalized to the range [0,1] by min–max method given as follows:

$$y_j = \frac{X_j - X_{\min}}{X_{\max} - X_{\min}}. \quad (13)$$

Here, y_j denotes the normalized data, X_j is the original data, $X_{\min}$ is the minimum of the original data, and $X_{\max}$ is the maximum of the original data.

80% of the data were selected for training and the remaining 20% were chosen for test. Test data were determined randomly. In addition, mean square error (MSE), root mean square error (RMSE), coefficient of determination (R^2), and mean absolute percentage error (MAPE) were used as performance indexes for methods.

$$MSE = \frac{1}{n}\sum_{j=1}^{n}\left(y_j - \hat{y}_j\right)^2 \quad (14)$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{j=1}^{n}\left(y_j - \hat{y}_j\right)^2} \quad (15)$$

$$R^2 = 1 - \frac{\sum_{j=1}^{n}\left(y_j - \hat{y}_j\right)^2}{\sum_{j=1}^{n}\left(y_j - \overline{y}\right)^2} \quad (16)$$

$$MAPE = \left(\frac{1}{n}\sum_{j=1}^{n}\frac{\left|y_j - \hat{y}_j\right|}{y_j}\right) \times 100 \quad (17)$$

Here, y_j denotes the actual output value, $\hat{y}_j$ indicates the predicted output value, $\overline{y}$ shows the mean of the actual output value, and n represents the number of samples.

Table 1 Air pollution descriptive statistics of İstanbul city for 2019

Air pollutants	Minimum	Maximum	Mean	Std. deviation
$PM_{2.5}$(ug/m^3)	25.00	157.00	62.8300	22.2116
PM_{10}(ug/m^3)	10.00	73.00	33.6629	12.3691
Wind speed	1.00	12.60	4.3227	1.9400
Wind gust	0.80	35.10	8.1042	5.2181
Temperature	0.00	28.50	16.4660	6.8755
SO_2 (μ g/m^3)	1.60	15.80	7.8725	3.7440
Pressure	996.50	1031.80	101.4439	5.8247
O_3 (ug/m^3)	1.50	41.60	19.6578	9.5832
NO_2 (μ g/m^3)	4.90	45.80	17.6516	7.2156
Humidity	35.00	94.50	69.8901	9.6001
CO (ug/m^3)	5.50	32.10	17.5575	5.8046

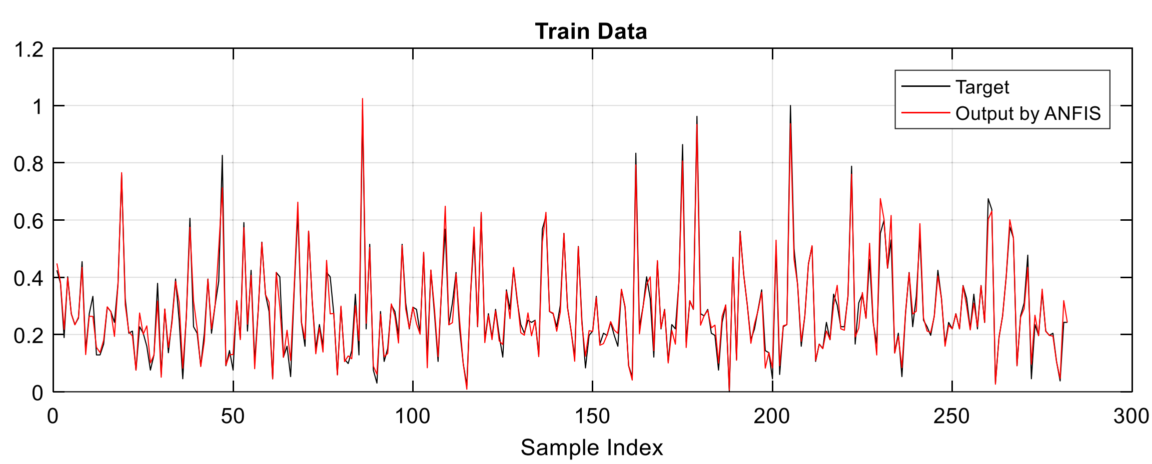

Fig. 2 The results of ANFIS for train data

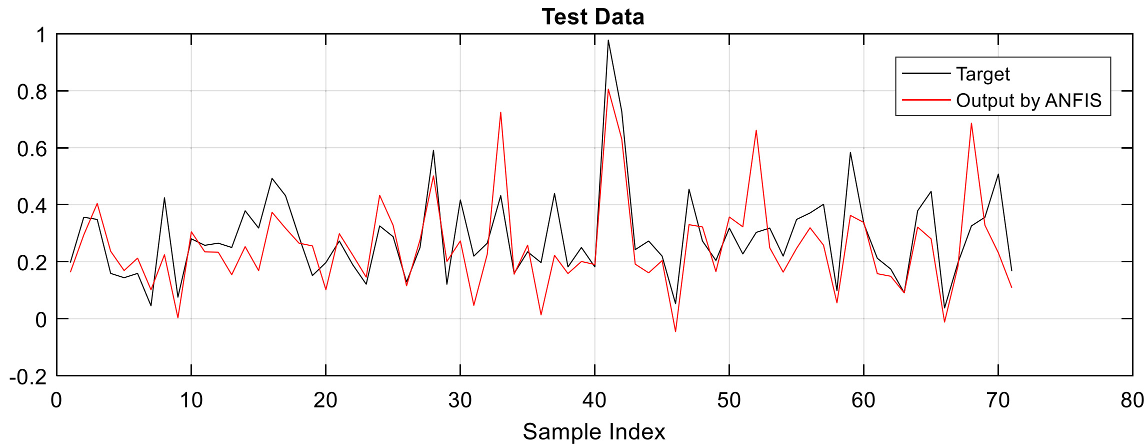

Fig. 3 The results of ANFIS for test data

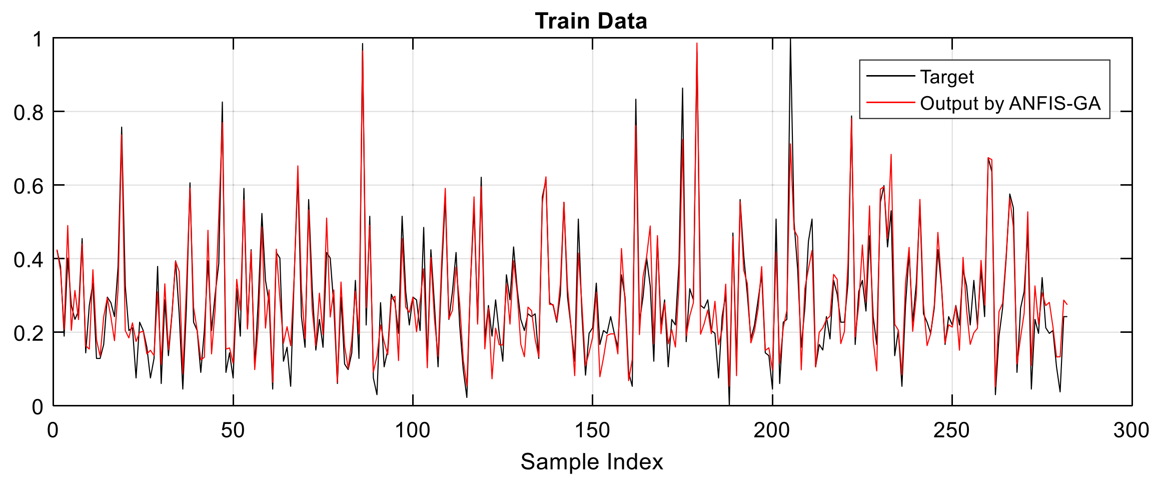

Fig. 4 The results of trained ANFIS by GA for train data

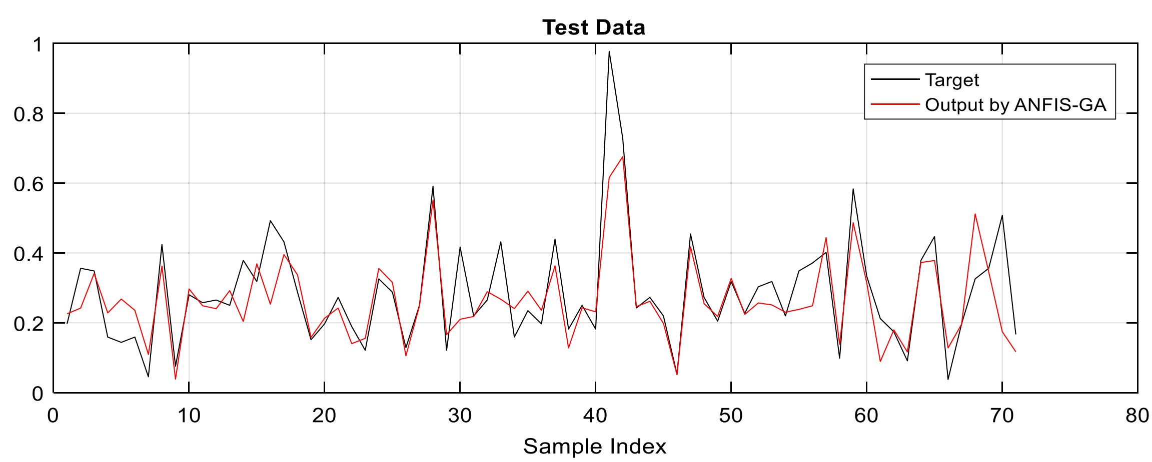

Fig. 5 The results of trained ANFIS by GA for test data

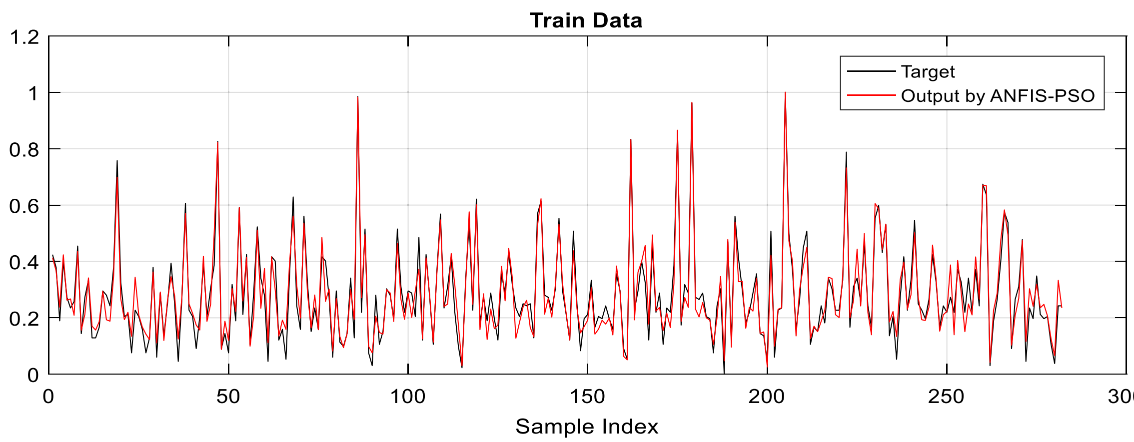

Fig. 6 The results of trained ANFIS by PSO for train data

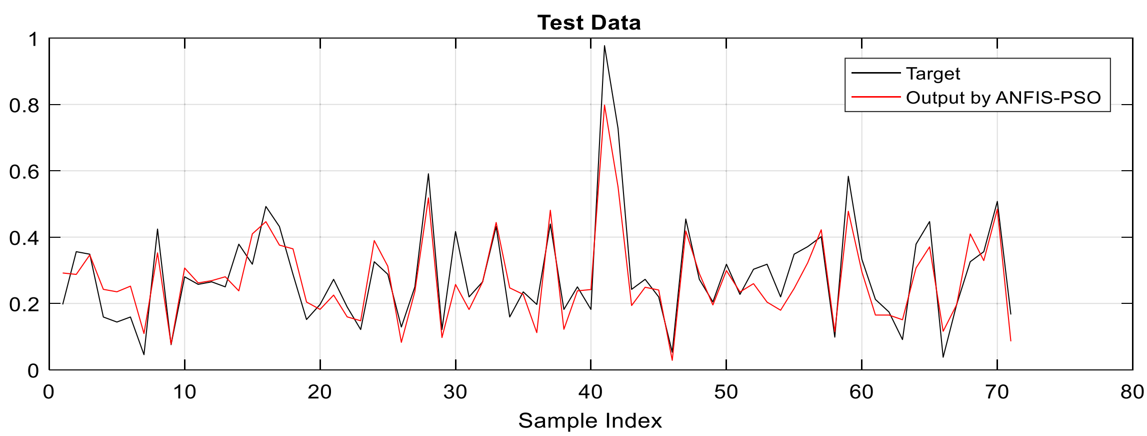

Fig. 7 The results of trained ANFIS by PSO for test data

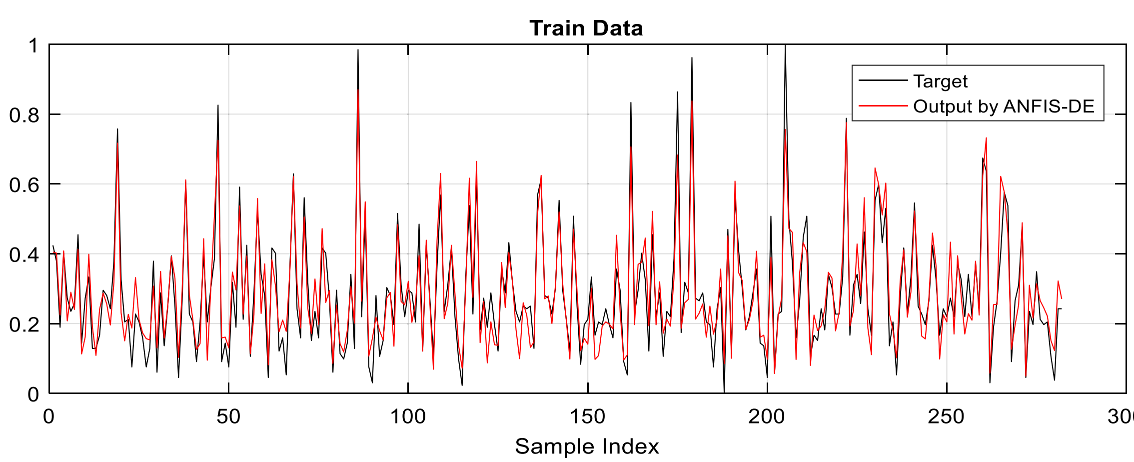

Fig. 8 The results of trained ANFIS by DE for train data

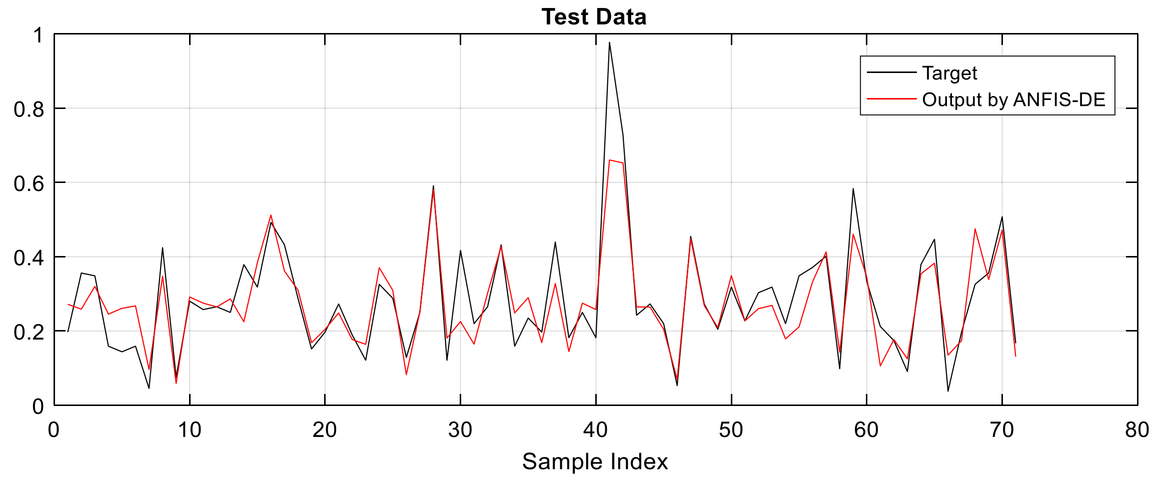

Fig. 9 The results of trained ANFIS by DE for test data

Table 2 Comparison of performance of methods

Methods	MSE	RMSE	R^2	MAPE
ANFIS	0.013763	0.11732	0.74753	35.0771
ANFIS-GA	0.0085764	0.092609	0.82174	25.3104
ANFIS-PSO	**0.0043461**	**0.065925**	**0.91608**	**24.2287**
ANFIS-DE	0.005373	0.073301	0.89127	24.3983

Figures 2, 3, 4, 5, 6, 7, 8, 9 demonstrate the training phase and testing phase results of ANFIS, trained ANFIS by GA, PSO, and DE to predict $PM_{2.5}$ values, respectively.

When we examine the prediction graphs of the test data, we can say that the DE and PSO-trained ANFIS structures give prediction results closer to the real values than the classical ANFIS structure. However, we cannot reach a clear conclusion about which algorithm is better just by looking at the graphs. Therefore, we use various performance indexes to compare the errors of these methods. Table 2 demonstrates the comparison results of ANFIS and trained ANFIS by various metaheuristic methods: GA, PSO, and DE for predicting the air pollutant $PM_{2.5}$ values. The best values of performance indexes show in bold font. This table shows that trained ANFIS structures are better than the classical ANFIS model with low MSE, RMSE, and MAPE values and high R^2. Furthermore, it is shown that the obtained results of trained ANFIS by PSO are better than both classic ANFIS and trained ANFIS structured by GA and DE with lowest MSE, RMSE, and MAPE, and highest R^2 in testing phases.

Discussion

Air, an excellent natural resource that helps people maintain their lives on this earth, is getting polluted by various human activities. Because of the several effects of air pollution, researchers tend to monitor air quality to reduce and control its severity. Many studies show the good efficiency of artificial methods for predictive analysis of air pollutant $PM_{2.5}$, one of the most important indicators of air pollution, using other pollutants and meteorological parameters (Ganesh 2018; Chen 2018). In this study, we used the ANFIS, one of the artificial intelligence methods, to predict $PM_{2.5}$ values for İstanbul.

In the classical ANFIS method, parameter tuning, also called training, is conducted by Gradient Descent and Least Square methods. However, that these methods get trapped in local optimal is a disadvantage for ANFIS. To eliminate this disadvantage and reach the global optimal, the use of heuristic methods in the training of ANFIS has recently become widespread. Evolutionary studies research shows remarkable advantages of GA, PSO, and DE (Wang et al. 2012; Sheniha et al. 2013; Rai et al. 2015; Chang et al. 2015; Baghban et al. 2016; Ghasemi et al. 2016) for the training of ANFIS. This study uses ANFIS models trained with GA, PSO, and DE to predict air pollutant $PM_{2.5}$ values.

It has been supported that the method used in training is critical to achieving practical results with ANFIS in this study. By hybridizing ANFIS with heuristic methods, better predictive values for air pollutant $PM_{2.5}$ has been obtained.

Conclusions

In this study, ANFIS is trained using the various popular metaheuristic algorithms, GA, PSO, and DE, to model and predict air pollutant $PM_{2.5}$ for İstanbul province. The prediction results obtained by the trained ANFIS with GA, PSO, and DE algorithms and classical ANFIS are compared using the prediction graphs and performance criteria. It has been observed that the trained ANFIS structures give better prediction values than classical ANFIS according to MSE, RMSE, R^2, and MAPE criteria. Furthermore, it seems that the performance of the PSO algorithm is better than other metaheuristic algorithms in ANFIS training for predicting air pollutant $PM_{2.5}$. As a result, it can be recommended to use trained ANFIS structures instead of the classical ANFIS method for such studies. As a future study, air pollution estimates for other provinces can be made using the methods in this study. In addition, ANFIS can be trained with different metaheuristic methods and compared with the results of this study.

References

Abbaspour-Gilandeh M, Abbaspour-Gilandeh Y (2019) Modelling soil compaction of agricultural soils using fuzzy logic approach and adaptive neuro-fuzzy inference system (ANFIS) approaches. Model Earth Syst Environ 5:13–20. https://doi.org/10.1007/s40808-018-0514-1

Afghan FR, Habib H, Akhunzada NA et al (2022) Customization of GIS for spatial and temporal analyses of air quality index trends in Kabul city. Model Earth Syst Environ. https://doi.org/10.1007/s40808-022-01396-5

Air Quality Open Data Platform (2021). https://aqicn.org/data-platform/covid19/verify/b8ddd06f-bbff-4e59-ba34-54f0af36b560. Accessed 21 Jan 2022

Amanollahi J, Ausati S (2020a) PM 2.5 concentration forecasting using ANFIS, EEMD-GRNN, MLP, and MLR models: a case study of Tehran. Iran Air Quality, Atmos Health 13:161–171

Amanollahi J, Ausati S (2020b) Validation of linear, nonlinear, and hybrid models for predicting particulate matter concentration in Tehran, Iran. Theor Appl Climatol. https://doi.org/10.1007/s00704-020-03115-5

Baghban A, Bahadori M, Ahmad Z, Kashiwao T, Bahadori A (2016) Modeling of true vapor pressure of petroleum products using ANFIS algorithm. Pet Sci Technol 34:933–939. https://doi.org/10.1080/10916466.2016.1170843

Barthwal A, Acharya D (2022) Performance analysis of sensing-based extreme value models for urban air pollution peaks. Model Earth Syst Environ. https://doi.org/10.1007/s40808-022-01349-y

Bhagowati B, Talukdar B, Narzary BK, Ahamad KU (2022) Prediction of lake eutrophication using ANN and ANFIS by artificial simulation of lake ecosystem. Model Earth Syst Environ. https://doi.org/10.1007/s40808-022-01377-8

Bhardwaj R, Pruthi D (2020) Evolutionary techniques for optimizing air quality model. Proced Comput Sci 167:1872–1879

Cekim HO (2020) Forecasting PM10 concentrations using time series models: a case of the most polluted cities in Turkey. Environ Sci Pollut Res 27:25612–25624

Chang BR, Tsai HF, Wang YA, Kuo CF (2015) Intelligent adaptation to in-cloud NoSQL database remote backup between data centers. In: ACM international conference proceeding series.

Chen Y (2018) Prediction algorithm of PM2.5 mass concentration based on adaptive BP neural network. Comput 100(8):825–838

Das S, Mullick SS, Suganthan PN (2016) Recent advances in differential evolution–an updated survey. Swarm Evolut Comput 27:1–30

Eberhart R, Kennedy J (1995) A new optimizer using particle swarm theory. Paper presented at the Proceedings of the Sixth International Symposium on Micro Machine and Human Science

Ganesh SS, Arulmozhivarman P, Tatavarti VSN (2018) Prediction of PM2.5 using an ensemble of artificial neural networks and regression models. Ambient Intel Human Comput. https://doi.org/10.1007/s12652-018-0801-8

Ghasemi A, Amanollahi J (2019) Integration of ANFIS model and forward selection method for air quality forecasting. Air Qual Atmos Health 12:59–72

Ghasemi E, Kalhori H, Bagherpour R (2016) A new hybrid ANFIS-PSO model for prediction of peak particle velocity due to bench blasting. Eng Comput 32:607–614. https://doi.org/10.1007/s00366-016-0438-1

Goldberg DE, Holland JH (1988) Genetic algorithms and machine learning. Mach Learn 3:95–99

Gui L, Xia X, Yu F, Wu H, Wu R, Wei B, Zhang Y, Li X, He G (2019) A multi-role based differential evolution. Swarm Evolut Comput 50:100508

Holland JH (1975) Adaptation in natural and artificial systems. The University of Michigan Press, Ann Arbor, p 1

Jang J-S (1993) ANFIS: adaptive-network-based fuzzy inference system. IEEE Trans Syst, Man, Cybernetics 23:665–685

Karaboga D, Kaya E (2019) Adaptive network based fuzzy inference system (ANFIS) training approaches: a comprehensive survey. Arti Intell Rev 52:2263–2293

Mirzaei M, Amanollahi J, Tzanis CG (2019) Evaluation of linear, nonlinear, and hybrid models for predicting PM 2.5 based on a GTWR model and MODIS AOD data. Air Qual Atmos Health 12:1215–1224

Ordóñez-De León B, Aceves-Fernandez M, Fernandez-Fraga S, Ramos-Arreguín J, Gorrostieta-Hurtado E (2020) An improved particle swarm optimization (PSO): method to enhance modeling of airborne particulate matter (PM10). Evol Syst 11:615–624

Pahlavani H, Dehghani AA, Bahremand AR, Shojaei S (2017) Intelligent estimation of flood hydrographs using an adaptive neuro–fuzzy inference system (ANFIS). Model Earth Syst Environ 3:1–9

Polat K (2012) A novel data preprocessing method to estimate the air pollution (SO 2): neighbor-based feature scaling (NBFS). Neural Comput Appl 21(8):1987–1994

Polat K, Durduran SS (2012) Usage of output-dependent data scaling in modeling and prediction of air pollution daily concentration values (PM 10) in the city of Konya. Neural Comput Appl 21:2153–2162

Price K, Storn RM, Lampinen JA (2006) Differential evolution: a practical approach to global optimization. Springer Science & Business Media, Germany

Purnomo MR, Anugerah AR (2020) Achieving sustainable environment through prediction of air pollutants in Yogyakarta using adaptive neuro fuzzy inference system. Eng Sci Tech 15:2995–3012

Rai AA, Pai PS, Rao BRS (2015) Prediction models for performance and emissions of a dual fuel CI engine using ANFIS. Sadhana Acad Proc Eng Sci 40:515–535. https://doi.org/10.1007/s12046-014-0320-z

Sheniha SF, Priyadharsini SS, Rajan SE (2013) Removal of artifact from EEG signal using differential evolution algorithm. In: International conference on communication and signal processing, ICCSP 2013. Proceedings, pp 134–138.

Shukura OB (2020) Using the MLR and neuro-fuzzy methods to forecast air pollution datasets. Int J Adv Sci Eng Inform Technol 10:1457–1464

Stanley Raj A, Oliver DH, Srinivas Y (2015) An automatic inversion tool for geoelectrical resistivity data using supervised learning algorithm of adaptive neuro fuzzy inference system (ANFIS). Model Earth Syst Environ 1:6. https://doi.org/10.1007/s40808-015-0006-5

Storn R (1996) On the usage of differential evolution for function optimization. Paper presented at the Fuzzy Information Processing Society, Biennial Conference of the North American

Storn R, Price K (1997) Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. Glob Opt 11:341–359

Talbi E-G (2009) Metaheuristics: from design to implementation. Wiley, Canada

Tauqir A, Kashif S (2022) COVID-19 outbreak and air quality of Lahore, Pakistan: evidence from asymmetric causality analysis. Model Earth Syst Environ 8:2115–2122

Tunckaya Y (2020) Performance analysis of novel air pollution forecasting system design in a Turkish cement plant via neural and neuro-fuzzy soft computing. Energy Sour Part A: Recov Utilization Environ Effects. https://doi.org/10.1080/15567036.2020.1825561

Wang R, Zhang J, Zhang Y, Wang X (2012) Assessment of human operator functional state using a novel differential evolution optimization based adaptive fuzzy model. Biomed Signal Process Control 7:490–498

Yalçınkaya A, Şenoğlu B, Yolcu U (2018) Maximum likelihood estimation for the parameters of skew normal distribution using genetic algorithm. Swarm Evolut Comp 38:127–138

Yang XS (2010) Engineering optimization: an introduction with metaheuristic applications. Wiley, Cambridge

Yang XS (2014) Nature-inspired optimization algorithms. Elsevier, London

Yilmaz M, Tosunoğlu F, Kaplan NH, Üneş F, Hanay YS (2022) Predicting monthly streamflow using artificial neural networks and wavelet neural networks models. Model Earth Syst Environ. https://doi.org/10.1007/s40808-022-01403-9

Zhou Y, Chang L-C, Chang F-J (2020) Explore a multivariate Bayesian uncertainty processor driven by artificial neural networks for probabilistic PM2.5 forecasting. Sci Total Environ 711:134792

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

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