Alexandria Engineering Journal 84 (2023) 184–203

Contents lists available at ScienceDirect

Alexandria Engineering Journal

journal homepage: www.elsevier.com/locate/aej

Original Article

Prediction of the thermal behavior of multi-walled carbon nanotubes-CuO-CeO_2 (20-40-40)/water hybrid nanofluid using different types of regressors and evolutionary algorithms for designing the best artificial neural network modeling

Reza Rostamzadeh-Renani [a], Mohammadreza Baghoolizadeh [b], S. Mohammad Sajadi [c], Mostafa Pirmoradian [d], Mohammad Rostamzadeh-Renani [a], Sh. Baghaei [d], Soheil Salahshour [e,f,g]

[a] Energy Department, Politecnico di Milano, Via Lambruschini 4, 20156 Milan, Italy
[b] Department of Mechanical Engineering, Shahrekord University, Shahrekord 88186-34141, Iran
[c] Department of Nutrition, Cihan University-Erbil, Kurdistan Region, Iraq
[d] Department of Mechanical Engineering, Khomeinishahr Branch, Islamic Azad University, Khomeinishahr, Iran
[e] Faculty of Engineering and Natural Sciences, Istanbul Okan University, Istanbul, Turkey
[f] Faculty of Engineering and Natural Sciences, Bahcesehir University, Istanbul, Turkey
[g] Department of Computer Science and Mathematics, Lebanese American University, Beirut, Lebanon

ARTICLE INFO

Keywords:
Thermal behavior
Hybrid nanofluid
Regressors
Evolutionary algorithms
Artificial neural network modeling

ABSTRACT

For conducting an analysis of the experimental data, it is imperative to establish a mathematical correlation between the input and output variables. This entails executing a curve fitting or regression procedure on the data, for which numerous methodologies exist. Within the scope of present investigation, the design variables encompass the solid volume fraction (φ) and temperature. Thermal conductivity (TC) of MWCNT-CuO-CeO_2 (20-40-40)/water hybrid nanofluid (HNF) is also the objective function. Ten different types of regressors are utilized for regression operations which are Multiple Linear Regression (MLR), Decision Tree (D-Tree), Multi-Layer Perceptron (MLP), Support Vector Machine (SVM), Extreme Learning Machine (ELM), Radial Basis Function (RBF), Adaptive Neuro-Fuzzy Inference System (ANFIS), Gaussian Process Regression (GPR), Multivariate Polynomial Regression (MPR) and Group Method of Data Handling (GMDH). Once the governing equations linking the design variables and the objective functions have been established, these equations can be employed to forecast the simulation data. By substituting the above input values into the equations, we can calculate the corresponding output values for the TC of the HNF. The results obtained from the MPR algorithm are compared to the experimental data. For the GPR, MLR, D-Tree, ELM, MPR, MLP, RBF, SVM, ANFIS, and GMDH algorithms, the maximum margin of error is found to be 0.031, 0.02579, 0.028946, 0.033889, 0.01568, 0.02515, 0.03485, 0.03, 0.0385, and 0.0178, respectively. Moreover, the kernel density estimation diagram indicates the gap between experimental data and data predicted by regression algorithms. Finally, it is evident that the MPR algorithm demonstrates to have a reduced residual dispersion, with the residuals approaching zero.

1. Introduction

Adding materials with a high thermal conductivity (TC) to the base fluid (BF) is one method for heat transfer enhancement in heat exchangers. Over the course of many years, researchers have devoted a substantial amount of time and energy to explore the possibility of enhancing heat transfer by employing mixtures of micron-sized solid particles suspended in fluids. However, these fluids had some issues including sedimentation, impurities, corrosion, and increased pressure loss, among others. Prior to 1881, when Maxwell first proposed the use of nano-sized particles, the field of heat transfer in fluids had not experienced a significant progression. However, this changed dramatically after 1881. He presented a novel perspective on the behavior of solid–fluid suspensions with nanoparticles. Masuda et al. [1] initially used the term "nanofluid (NF)" to describe a fluid that contained suspended particles. Choi [2] then expanded this concept significantly while working at the American Argonne Laboratory. NF is a suspension of very fine solid particles with dimensions between 1 and 100 nm. Nanoparticles (NPs) are significantly more stable than larger particles like microparticles and have a larger contact surface area with the fluid region. The two primary characteristics of NF are its extremely high stability and TC. BFs are typically lower conductive fluids such as water, ethylene glycol (EG), and fluids of this category that are employed for

https://doi.org/10.1016/j.aej.2023.10.059
Received 16 May 2023; Received in revised form 14 October 2023; Accepted 29 October 2023
Available online 10 November 2023

Nomenclature

ANFIS	Adaptive Neuro-Fuzzy Inference System
BF	base fluid
R	Correlation Coefficient
D-Tree	Decision Tree
EG	ethylene glycol
ELM	Extreme Learning Machine
GA	Genetic Algorithm
GMDH	Group Method of Data Handling
GPR	Gaussian Process Regression
HNF	hybrid nanofluid
KDE	Kernel Density Estimation
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MLP	Multi-Layer Perceptron
MLR	Multiple Linear Regression
MOD	Margin of Deviation
MPR	Multivariate Polynomial Regression
MSE	Mean Squared Error
NF	nanofluid
NFE	Number of Function Evaluations
NPs	Nanoparticles
RBF	Radial Basis Function
RMSD	Root Mean Square Deviation
RMSE	Root Mean Squared Error
R^2	R squared
φ	solid volume fraction
SVM	Support Vector Machine
T	Temperature
TC	Thermal Conductivity

heat exchange in industry, while NPs are typically formed from metals such as Cu, Al, Potassium, Fe, and their oxides, and carbon nanotubes. Due to the minute size of the particles, the feasibility of corrosion and pressure loss are substantially reduced, and the fluids' resistance to sedimentation is significantly increased. In the past decade, numerous studies on the NFs were conducted [3–5]. Due to the presence of multiple effective parameters in the TC and the fact that the number and circumstances of experiments and research have not been proportional to the enormous number of NFs, no exhaustive model or relationship has yet been developed in this field. Recent researches have not only been unable to provide an accurate model estimate, but they have also been unable to adequately explain and justify the disparities and even inconsistencies that have been identified among the results of numerous experiments. The TC of NFs has been found to be influenced by various factors, including the Solid Volume Fraction (φ) of NPs, the characteristics of suspended particles and BF, the shape and size of NPs, the presence of surfactants and stabilizers, the choice of ultrasonic stirrer, the duration of the stirring process, and the adhesion time and accumulation of particles. On the other hand, the mutual influence of each of these characteristics on the others is one of the most complicated and novel topics of discussion among scholars. Numerous studies have been conducted on NFs to examine the impact of φ and temperature on their thermal properties. In the experiment that Lee et al. [6] conducted, NPs of CuO and Al_2O_3 were combined with EG to produce NFs. The researchers found that the TC of NFs containing even a trace number of NPs were much higher than those of the same liquids that did not contain any NPs at all. The researchers observed that the TC of NFs with even minimal amounts of NPs was much greater than the TC of identical liquids without any NPs present. TC increases linearly with increasing φ, according to the researchers' findings. In a study conducted by Xie et al. [7] that focused on the φ and shape of NPs, they added silicon carbide NPs to water and EG in two different shapes: spherical and cylindrical. They found that the cylindrically-shaped NPs increased the TC coefficient more than the spherically-shaped NPs did. They discovered that the coefficient of TC for nanofluids comprised of cylindrical NPs is approximately 11% greater than that of spherical NPs. Murshed et al. [5] investigated the effects of various CuO concentrations in water, ranging from $\varphi = 1.5\%$ to 5%. They noticed a nonlinear relationship between the TC and φ, particularly at low φ s. Vărdaru et al. [8] conducted an experimental study on the TC of a graphene oxide, silicon, and water-based hybrid NF. Furthermore, they investigated the effects of temperature (25–50 °C) and NP blending ratio on TC. Researchers have demonstrated that a rise in temperature of graphene oxide content increases the TC. Bakhtiari et al. [9] studied the TC of TiO_2-graphene/water hybrid NF at various φ s (0.005–0.5%) and temperatures (25–75 °C). Based on their empirical findings, it was observed that the enhancement of both temperature and φ leads to an improvement in the thermal conductivity of NFs. The findings indicate that a higher proportion of φ leads to a more significant boost in TC, particularly at higher temperatures. However, the effect of φ increase on TC was greater than that of temperature. At$\varphi = 0.5\%$ and T = 75 °C, the TC was 27.84% greater than that of the BF. Hemmat et al. [10] carried out an investigation on the relative TC of a hybrid NF made up of MWCNTs, MgO, and ZnO (15:50:35) NPs, together with water. It was shown that parameters such as temperature and φ affect the relative TC of the HNF. In the optimal testing scenario, a relative TC of 32.70% was determined. By

Table 1
Specifications of input data.

Number	φ (%)	Temperature (°C)	TC $(W/m.K)$	Number	φ (%)	Temperature (°C)	TC $(W/m.K)$
1	0	20	0.587	16	0.3	20	0.68
2	0	30	0.606	17	0.3	30	0.712
3	0	40	0.621	18	0.3	40	0.723
4	0	50	0.636	19	0.3	50	0.731
5	0	60	0.648	20	0.3	60	0.752
6	0.1	20	0.629	21	0.4	20	0.72
7	0.1	30	0.634	22	0.4	30	0.729
8	0.1	40	0.646	23	0.4	40	0.749
9	0.1	50	0.664	24	0.4	50	0.774
10	0.1	60	0.685	25	0.4	60	0.789
11	0.2	20	0.651	26	0.5	20	0.751
12	0.2	30	0.657	27	0.5	30	0.762
13	0.2	40	0.672	28	0.5	40	0.77
14	0.2	50	0.708	29	0.5	50	0.798
15	0.2	60	0.739	30	0.5	60	0.821

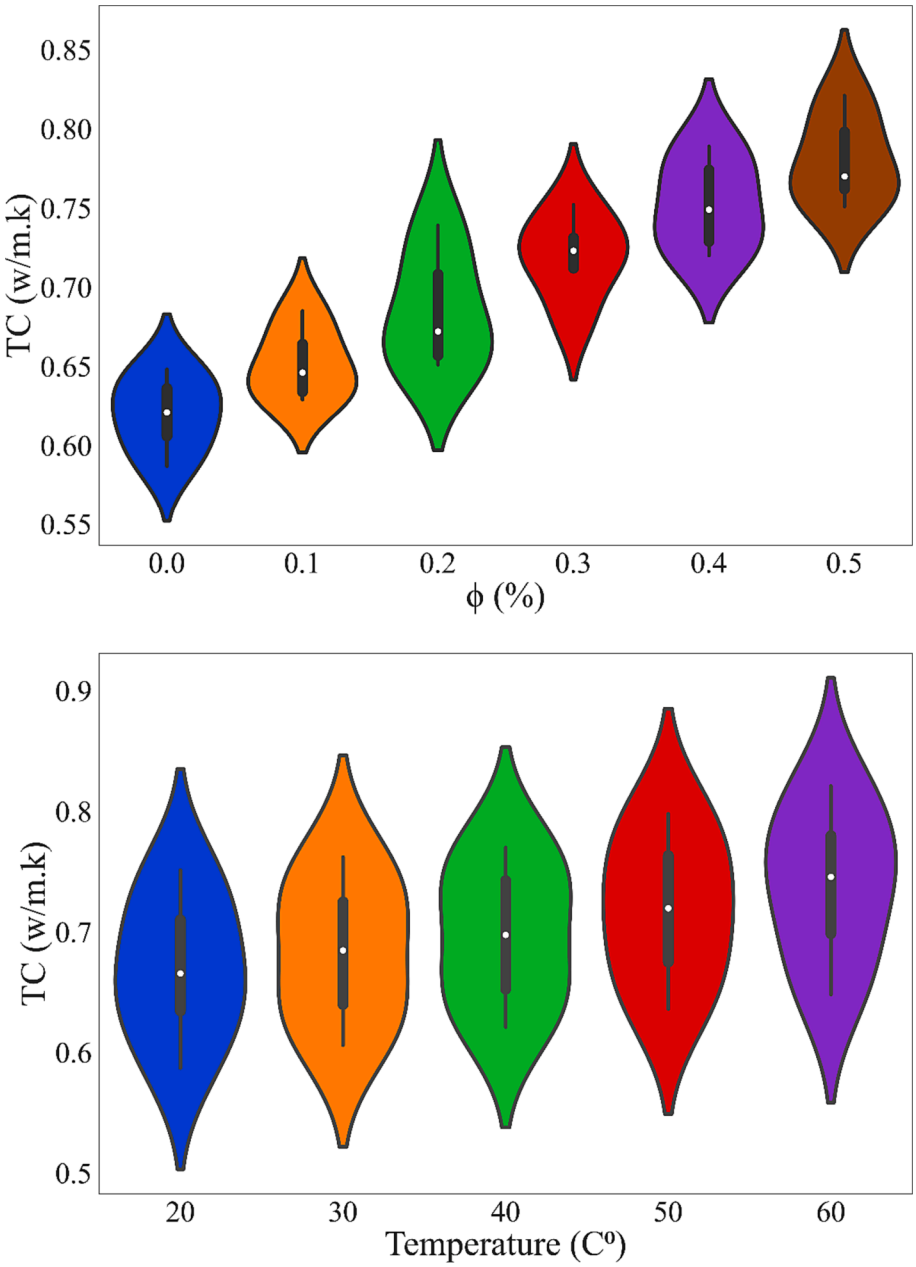


Fig. 1. The behavior of inputs on outputs with a violin plot.

Table 2
Sensitivity analysis to assess the contribution and influence of inputs on outputs.

Input variables	Conductivity	
	μ	σ
φ	36.0333	17.58
T	15.8	4.2

stirring SiO_2 and CuO NPs in an EG-based coolant, Kumar et al. [11] created a hybrid NF. Simulations were run with varying φ s (from 0.1 to 0.5%) to determine its thermophysical parameters and thermal characteristics. Overall, the TC increased by 6.5% when compared to water and EG coolant. Recent research findings suggest that temperature has a substantial role in influencing the thermal characteristics of NFs. However, prior investigations have not extensively explored the influence of temperature on the alteration of the TC of NFs. The TC of CuO/water and Al_2O_3/water NFs was studied by Li and Peterson [12]. At higher temperatures, they observed a more pronounced influence of φ for NF-containing aluminum oxide NPs. Karthik et al. [13] conducted a research on water-CuO NF to determine its TC. The findings of their investigation show that the TC of NF undergoes considerable shifts as the temperature varies. The study conducted by Fedela et al. [14] examined the behavior of titanium oxide NPs in aqueous solutions. The researchers found that the TC of the NF increases as both the φ and the temperature of the nanofluid increase. Gao et al. [15] employed molecular dynamics simulations to investigate the thermal performance and hydrodynamics of a NF that consisted of water, carbon nanotubes, and graphene nanosheets. They examined how the hydrodynamic and thermodynamic properties of the NF vary with temperature and pressure. The results demonstrated that a higher initial temperature improved oscillation amplitude and particle mobility, resulting in a higher TC. Adun et al. [16] studied the dynamic viscosity and TC of an Al_2O_3-ZnO-Fe_3O_4 hybrid NF with varying mixture ratios at various φ s

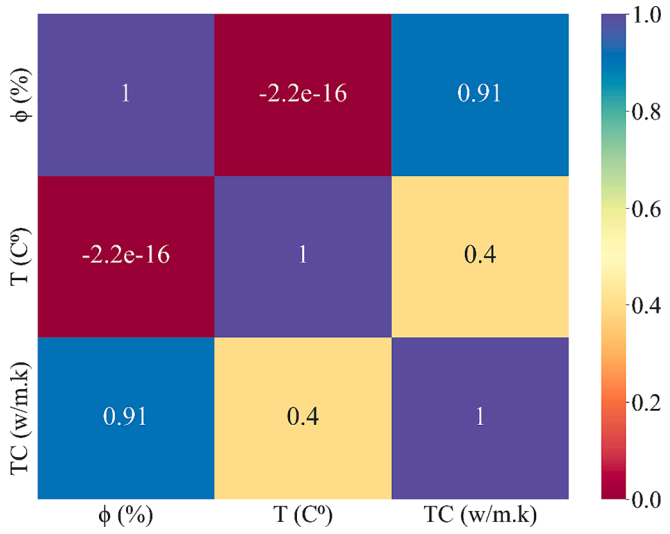


Fig. 2. Input and output heatmap diagram.

Table 3
Variables and range of changes in machine learning algorithms.

Algorithm	Variables and range			
MLP	Variables	Number of neurons	Number of layers	Activation function
	Range	[1,20]	[1,5]	[Tansig, Purelin]
GMDH	Variables	Number of neurons	Number of layers	Coefficient pressure
	Range	[1,5]	[1,5]	[0,1]
ELM	Variables	Number of neurons	Number of layers	–
	Range	[1,20]	[1,5]	–
SVM	Variables	Activation function	–	–
	Range	[Linear, Polynomial, RBF]	–	–
RBF	Variables	Number of neurons	Spread	–
	Range	[1,20]	[0,20]	–
D-Tree	Variables	Minimum Parent Size	–	–
	Range	[1,10]	–	–
ANFIS	Variables	Type function	Membership function	
	Range	sugeno	[1,10]	

for a range of temperatures (25 °C–65 °C). The outcome demonstrated that the NF's thermophysical characteristics were considerably influenced by temperature and φ. The mixture ratios 1:2:1 and 2:1:1 produce TC enhancements of 31.68% and 32.92%, respectively, while the 1:1:1 ternary HNF produces the maximum TC enhancement (36,018%). Atφ = 1%, the optimal TC enhancement (relative to water) was reported to be 18.98%, 28.588%, and 32.455% for mono, hybrid NF, and ternary NF, respectively. The TC of a hydraulic oil-based NF was studied by Shahsavar et al. [17] throughout a wide variety of φ s, temperatures, and mixing ratios in the presence of ternary nano-additives, graphene oxide, iron oxide, and titanium dioxide. The measurements showed that a rise in temperature and φ s enhanced the TC of ternary HNFs for all mixing ratios. The optimal conditions for TC augmentation were found to be at 65 °C and 1% VF. Guan et al. [18] used molecular dynamics simulation to investigate the mechanisms by which HNFs enhance TC. The microscopic principles behind the enhanced TC of HNFs were studied and discussed. The researchers conducted an estimation of the TC of both Cu and Ag nanofluids, as well as the Cu-Ag hybrid nanofluid. Their objective was to examine the influence of the hybridization ratio and identify the fundamental mechanisms responsible for the enhancement in the TC of HNFs.

In recent years, there has been a notable acceleration in the inclination towards employing modeling techniques as opposed to depending solely on statistical methods. The utilization of these methodologies has demonstrated a significant level of precision in predicting the results of many processes, mostly attributed to their direct reliance on empirical data. Several examples of such methodologies include genetic algorithms, adaptive neural-fuzzy inference systems, fuzzy logic, and artificial neural networks. Researchers in many different scientific and technological areas have taken an interest in Machine Learning because of its high response accuracy and generalizability to different settings [19–22]. In this way, Rostami et al. [23] used an ANN model and curve fitting to predict the TC of a hybrid NF made of graphene oxide and CuO, based on experimental data. From the curve-fitting method, they proposed a two-variable empirical correlation as a function of temperature and φ. The desired outcome in their study was the TC ratio, with temperature and φ serving as inputs for the ANN. Their results showed that the TC of the HNF could be accurately predicted using the ANN and the proposed correlation. By using an ANN model, Yang et al. [24] calculated the TC of WO_3-MWCNTs/ engine oil hybrid NF. The initial tests were performed at T = 20 °C to 60 °C withφ = 0.05 to 0.6%. The absolute error values of the ANN technique were found to be zero in many cases. Moreover, the ANN exhibited decreased error levels compared to the correlation technique. To generate a hybrid NF, Pare and Ghosh [25] dissolved ZnO, CuO, and alumina NPs in water. The study examined a range of temperatures spanning from 20 °C to 90 °C, as well as a range of φ values between 0.02% and 2%. The ANN model considered the φ and temperature as the input data, and the output data was TC of NF. The results of the ANN model matched those of the experiments. In a separate study, Hu al. [26] studied mechanism, application and prospects of nanoparticle -enhanced coolants in machining. To foretell the TC of silver/water NF, Ramezanizadeh and Nazari [27] suggested a correlation and an ANN model. The input data included particle size, φ, and temperature. After comparing their respective models, it became clear that the ANNs provided more accurate and relevant outcomes. PVR et al. [28] tested the TC and viscosity of a novel HNF at varying temperatures, using four different mixing ratios and three different volume concentrations. Nanofluid was prepared using copper oxide, graphene

Table 4
The value of evaluation criteria for the objective function TC.

Algorithm	R	RMSE	MAE	MSE	R^2	MAPE
MLP	0.99434	0.006601	0.005321	4.356*e−5	0.98872	0.777235
SVM	0.99431	0.006573	0.004096	4.032*e−5	0.98865	0.584963
RBF	0.98974	0.008821	0.005457	7.7809*e−5	0.97958	0.768879
D-Tree	0.99376	0.006825	0.0051	4.658*e−5	0.98756	0.71911
ELM	0.99495	0.0068	0.004931	4.6246*e−5	0.98993	0.70904
MLR	0.99356	0.006933	0.005333	4.807*e−5	0.987	0.761308
ANFIS	0.98208	0.011834	0.004943	14*e−5	0.98843	0.734157
GMDH	0.99463	0.00635	0.005126	4.032*e−5	0.9893	0.742135
MPR	0.99711	0.004654	0.003441	2.1657*e−5	0.99422	0.483951
GPR	0.99463	0.00633	0.004932	4.0066*e−5	0.9893	0.710336

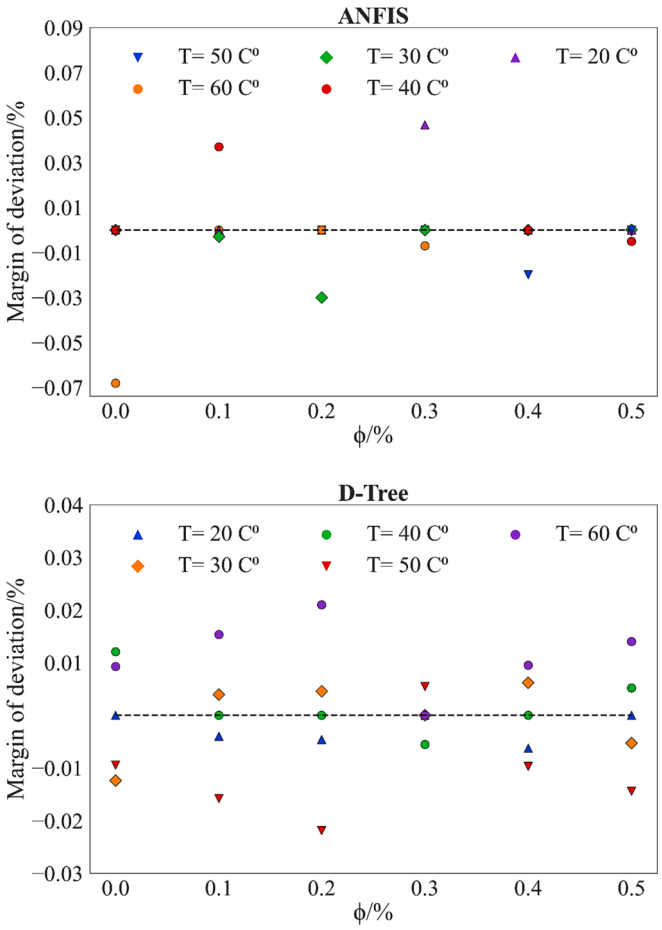


Fig. 3. Margin of deviation for TC of MWCNT-CuO-CeO_2 (20-40-40)/water HNF.

nanoplatelets, and alumina nanoparticles with a water-ethylene glycol mixture as the BF. The TC results demonstrated that graphene nanoplatelets dominate the particle mixture. Furthermore, an ANN model was developed to predict the TC and viscosity. A regression coefficient of 0.994996 was obtained, indicating that the ANN model and the extant experimental data were in good agreement. To determine how adding MWCNT to paraffin changed its TC, Rostami et al. [29] used ANN modeling and a response surface methodology (RSM). As input factors, φ (0.005–5%) and temperature (25–70 °C) were investigated, with the nanofluid TC as the output parameter. Calculations demonstrated that the ANN technique was more precise than the RSM. Colak [30] measured the TC of water-based ZrO_2 NFs at different φ s and temperatures ranging from 10 °C to 65 °C. Using the given experimental data, an ANN was constructed, and a new correlation for calculating TC values was proposed. The findings of the study indicated that the TC of NF was higher compared to that of BF. Furthermore, it was observed that the TC of NF rose as the φ and temperature increased. In addition, the results demonstrated that the created ANN and the correlation were in perfect accordance with the experimental data.

In this particular study, a total of ten different types of regressors, namely MLR, D-Tree, MLP, SVM, ELM, RBF, ANFIS, GPR, MPR, and GMDH, are employed to make predictions on the TC of a hybrid NF consisting of MWCNT-CuO-CeO_2 (20-40-40) dispersed in water. The design elements in present study are temperature and the solid volume fraction (φ). The dimensions of MWCNT, CuO and CeO_2 nanoparticles are 5–10 nm, 40–50 nm and 10–30 nm respectively. By employing the established governing equations that establish a relationship between the design variables and the objective functions, it becomes possible to make a prognostication on the outcomes of the simulation.

2. Description before processing

Currently, nanofluids are widely employed in various scenarios and contexts, making significant contributions to the fields of scientific research and industrial practices. Given the experimental nature of the research, conducting studies on them in a laboratory setting incurs significant costs and requires a substantial investment of time. Nevertheless, the implementation of robust artificial intelligence systems has the potential to mitigate the aforementioned testing . The field of data analysis encompasses a wide range of methodologies and techniques for prediction. Continuous advances and additions to this repertoire are observed on a daily basis. These methods can be classified as subgroups within the field of machine learning. The approaches in question encompass a range of mathematical techniques, including algorithms, linear equations, and polynomials involving one or more variables. Nevertheless, the aforementioned techniques may not possess the same level of simplicity as supervised algorithms and other neural networks, since they inherently exhibit a certain degree of intricacy. There are multiple approaches available for evaluating and optimizing performance. This paper will commence by analyzing the trajectory of the present investigation, followed by a comprehensive examination of the algorithms and methodologies employed in this research. This study will commence by introducing and afterwards discussing machine learning algorithms. Next, an analysis is conducted on the hyperparameters of these algorithms and their corresponding evaluation methods. Ultimately, the optimal fluid properties will be determined by the utilization

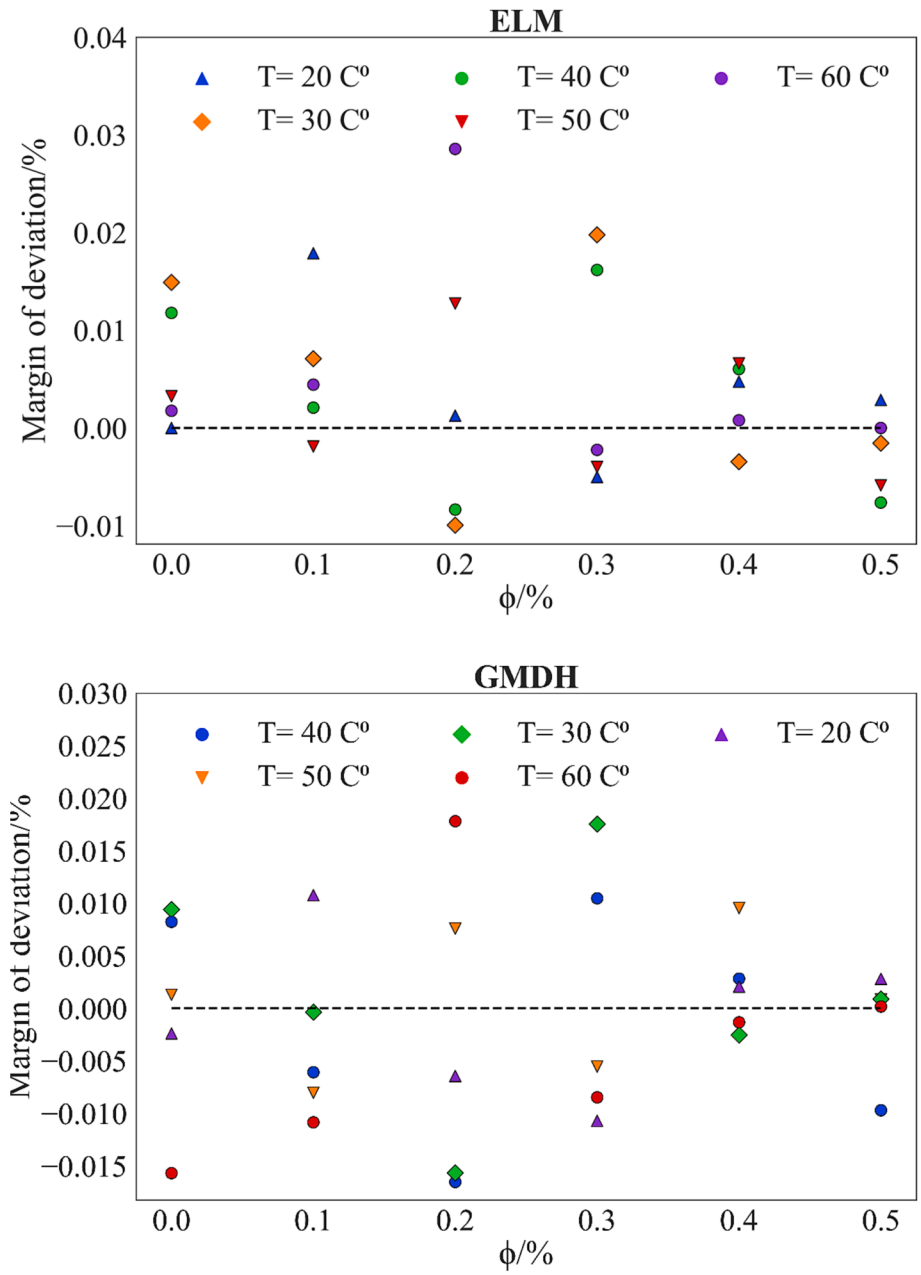


Fig. 3. (*continued*).

of the genetic algorithm.

2.1. Introduction of machine learning algorithms

To conduct a comprehensive analysis of the experimental data, it is imperative to establish a mathematical correlation between the input and output variables. This entails conducting a curve fitting or regression procedure on the data, utilizing a variety of methodologies. Design variables in this study are solid volume fraction (φ), and temperature. The objective function is also TC of MWCNT-CuO-CeO_2 (20-40-40)/water HNF. ANNs will be discussed later. In the current research, a total of ten regressors are employed for the purpose of conducting regression operations. These regressors include MLR, D-Tree, MLP, SVM, ELM, RBF, ANFIS, GPR, MPR, and GMDH. In the subsequent section, these algorithms will be shown.

2.1.1. Group method of data handling

This type of artificial neural network (ANN) has been employed by engineers in a variety of disciplines [31–37]. The primary objective of the GMDH ANN is to generate and predict meaningful output data. The input layer, the hidden layer, and the output layer are the three fundamental layers that make up this ANN [38,39]. Each neuron carries a polynomial equation because the inputs are gathered in the form of $\binom{n}{2}$ combinations. The neurons are combined similarly to reach the final layer, also known as the output layer. The input function and the output value of the GMDH ANN are, respectively, x and y. The approximate function f is the estimated output value that is predicted by ANN using $\hat{y}$ and also$\hat{f}$. In functions with multiple inputs M and one output, the actual values are calculated based on Eq. (1).

$$y = f(x_{i1}, x_{i2}, x_{i3}, \cdots, x_{in})(i = 1, 2, 3, \cdots, M) \quad (1)$$

With the input vector $X = (x_{i1}, x_{i2}, x_{i3}, \cdots, x_{in})$, the values of $\hat{y}$ are predicted by the function $\hat{f}$, and its equation is tested by Eq. (2).

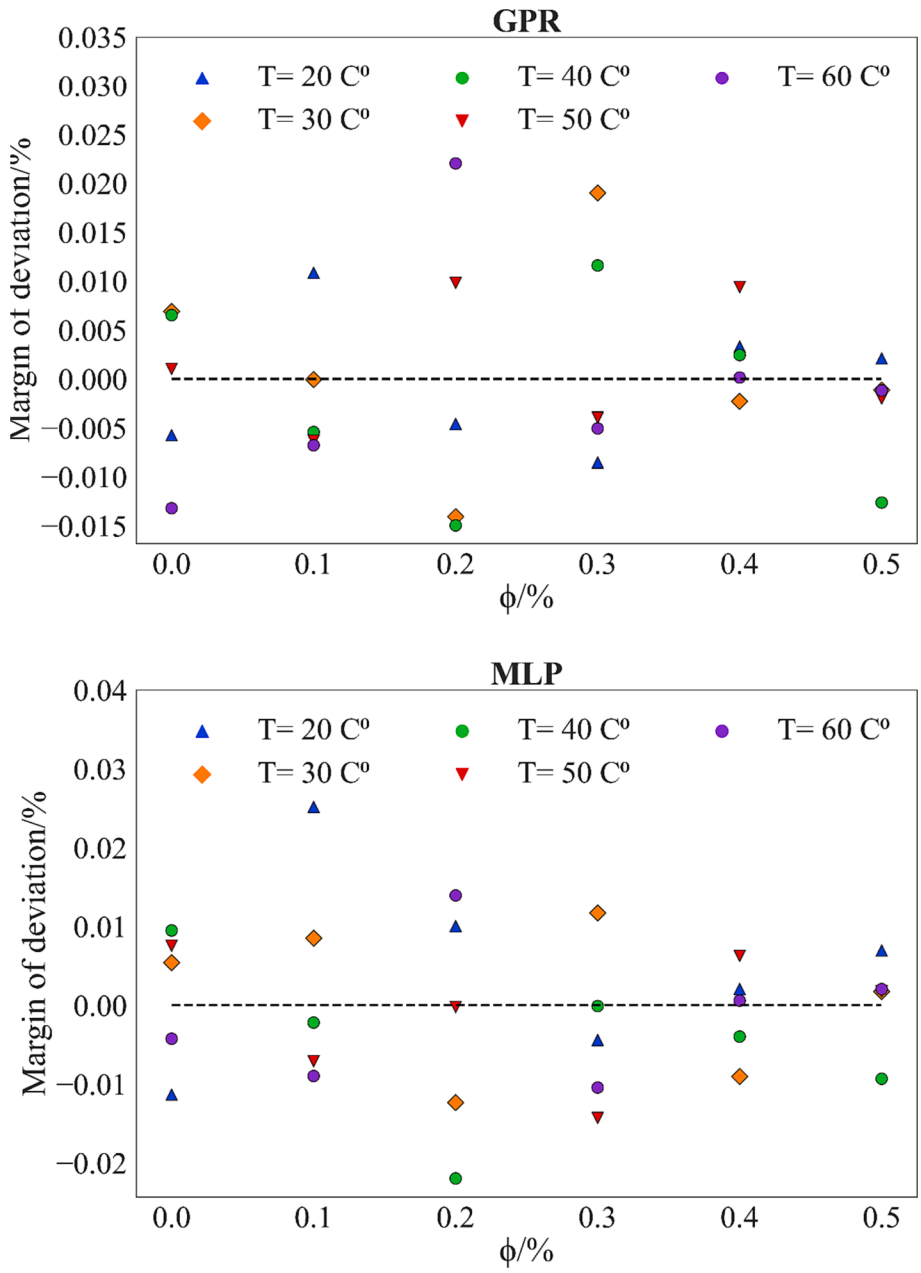


Fig. 3. (*continued*).

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

In the subsequent phase, the error squared value that must be minimized between the output and the estimated output of GMDH must be calculated. Using Eq. (3), the least squares error equation between these two outputs is estimated.

$$\sum_{i=1}^{M} [\widehat{f}(x_{i1}, x_{i2}, x_{i3}, \cdots., x_{in}) - \widehat{y}_i]^2 \rightarrow \min \quad (3)$$

The Volterra series connects the inputs and outputs ; whose relationship is given by Eq. (4).

$$y = a_0 + \sum_{i=1}^{n} a_i x_i + \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} x_i x_j + \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} a_{ijk} x_i x_j x_{k+} \cdots. \quad (4)$$

As shown in Eq. (5), the Kolmogorov-Gabor polynomials organize the polynomials discussed in Eq. (4).

$$\hat{y} = G(x_i, x_j) = a_0 + a_1 x_i + a_2 x_j + a_3 {x_i}^2 + a_4 {x_j}^2 + a_5 x_i x_j \quad (5)$$

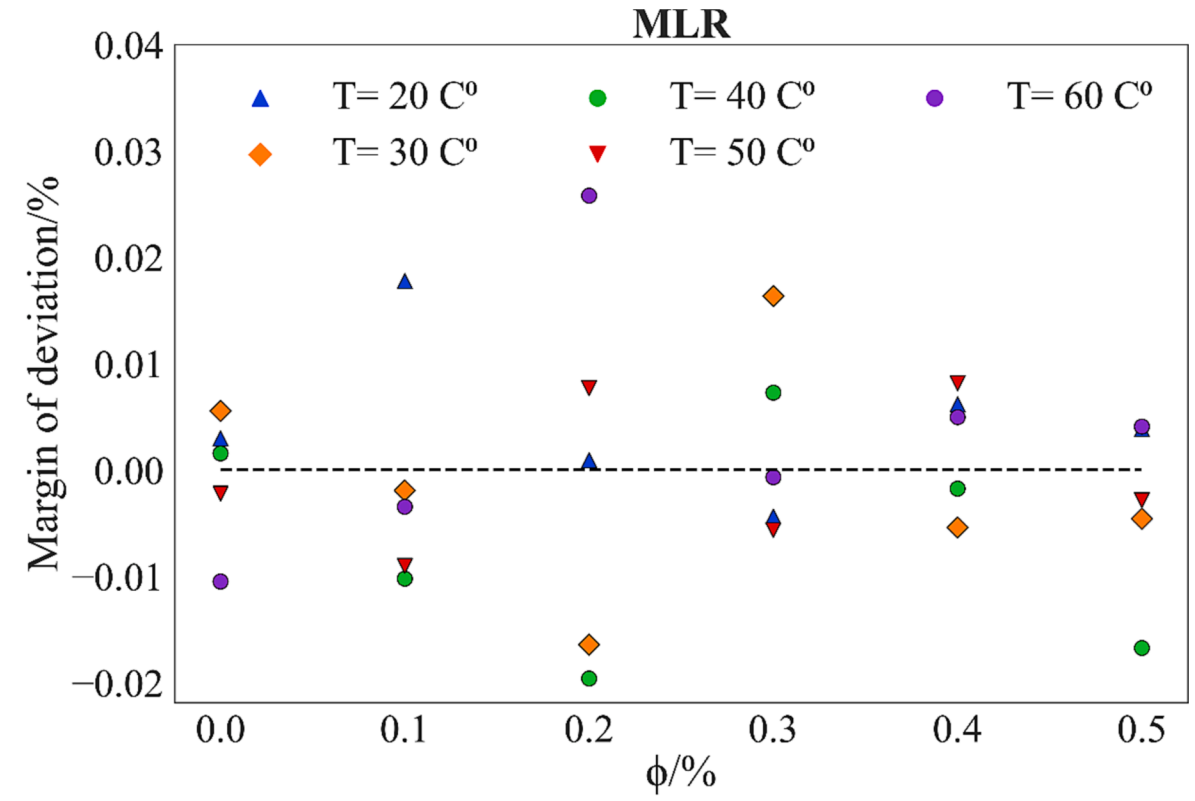
Regression is used to calculate the a_i coefficients, and Eqs. (1) and (2) are explained by Eq. (6), which must be minimized.

$$E = \frac{\sum_i^M (y_i - G_i)^2}{M} \rightarrow \min \quad (6)$$

The least squares method and the polynomial regression given by Eq. (2) can be used to predict the best outcome.

2.1.2. Multiple linear regression

Multiple simple regression (MLR) depicts the relationship between a group of independent variables and dependent variables. If there are independent variables $x_1, x_2, \ldots, x_n$, then to establish a linear relationship between them and the variable y, Eq. (7) must be developed between independent variables.



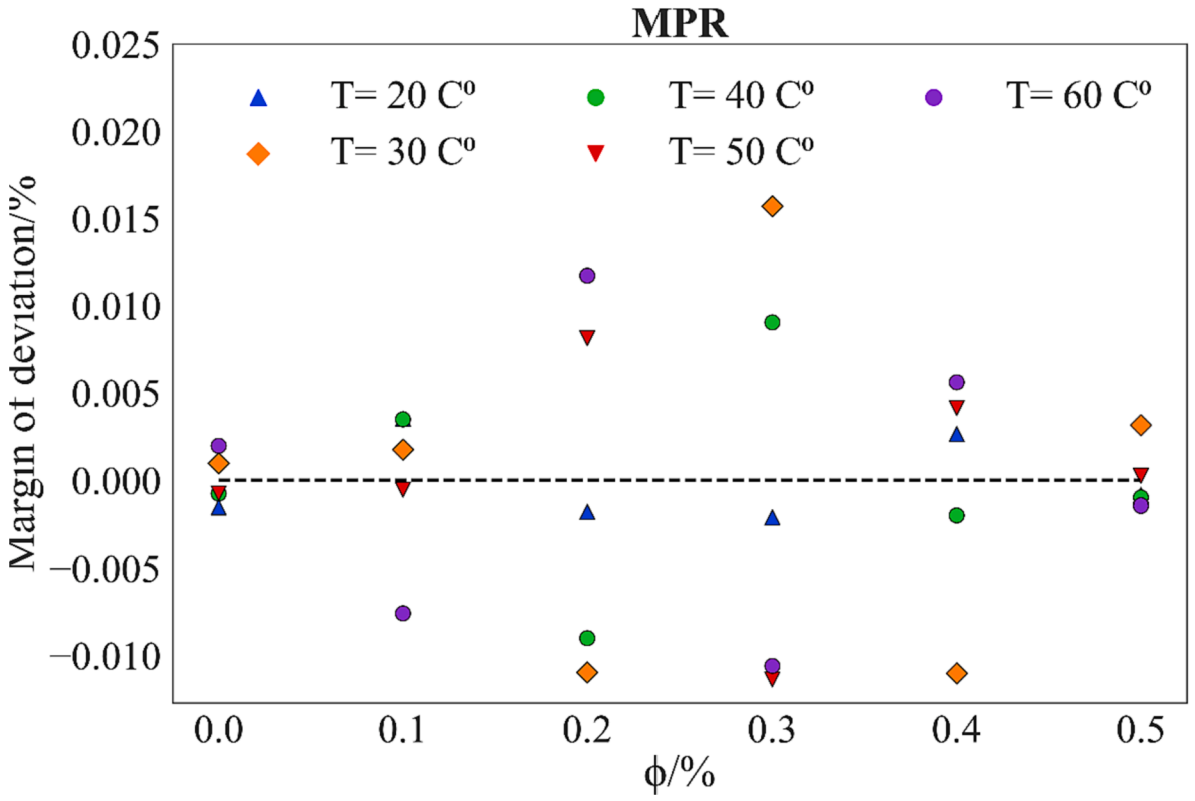


Fig. 3. (*continued*).

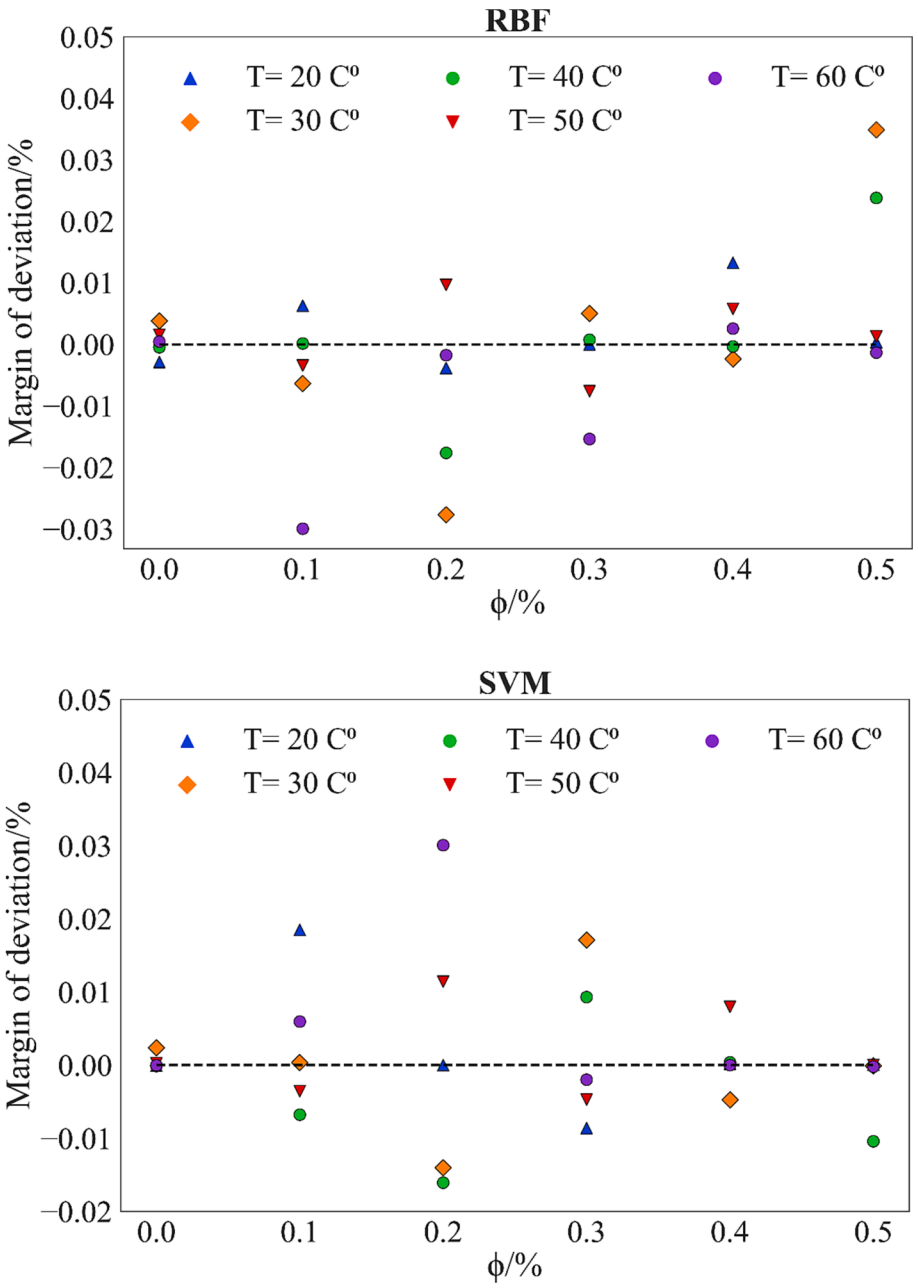


Fig. 3. (*continued*).

$$Y = a_0 + a_1x_1 + a_2x_2 + \cdots + a_nx_n \tag{7}$$

In this respect, the $a_1, a_2, \ldots, a_n$ values are called regression coefficients, and a_0 is the error value between the original and predicted variables.

2.1.3. D-tree

D-Tree algorithms are inspired by real trees. It comprises, by definition, of a root that branches out until it reaches a leaf. This algorithm's applications include the classification of discrete and continuous data and regression operations involving real or continuous input data [40]. The criterion for data segmentation is subset minimization. This algorithm can perform well with complex data, but as the data grows, so do the algorithm's branches and leaves, execution time is increased. To address this issue, a method to reduce the number of leaves was incorporated into the algorithm. Among these techniques, emphasis can be placed on pruning branches and determining the number of branch divisions. The classification of data by D in this algorithm is defined by Eq. (8).

$$D = (x_1, y_2), \ldots, (x_i, y_i), \ldots (x_n, y_n) \quad y_i \in R \quad x_i \in R^d \tag{8}$$

A D-Tree attempts to partition the data recursively so that each node has the same independent variable y. Each node is a subset of the recursively constructed data. Specifically, at node m, if the data is Q, select the dimension of the dependent variable and a threshold, and then divide the data in half based on this dimension and threshold, so that the mean becomes the independent variable or y becomes the independent variable. This dimension and threshold are called $\theta = (j, t_m)$. The range of j is equal to $[1, \cdots, d]$ and t_m is an integer. According to $\theta = (j, t_m)$, Q is divided into two sections $Q_{left}(\theta)$, and $Q_{right}(\theta)$, as explained in Eqs. (9)–(12).

$$N_m = |Q| \tag{9}$$

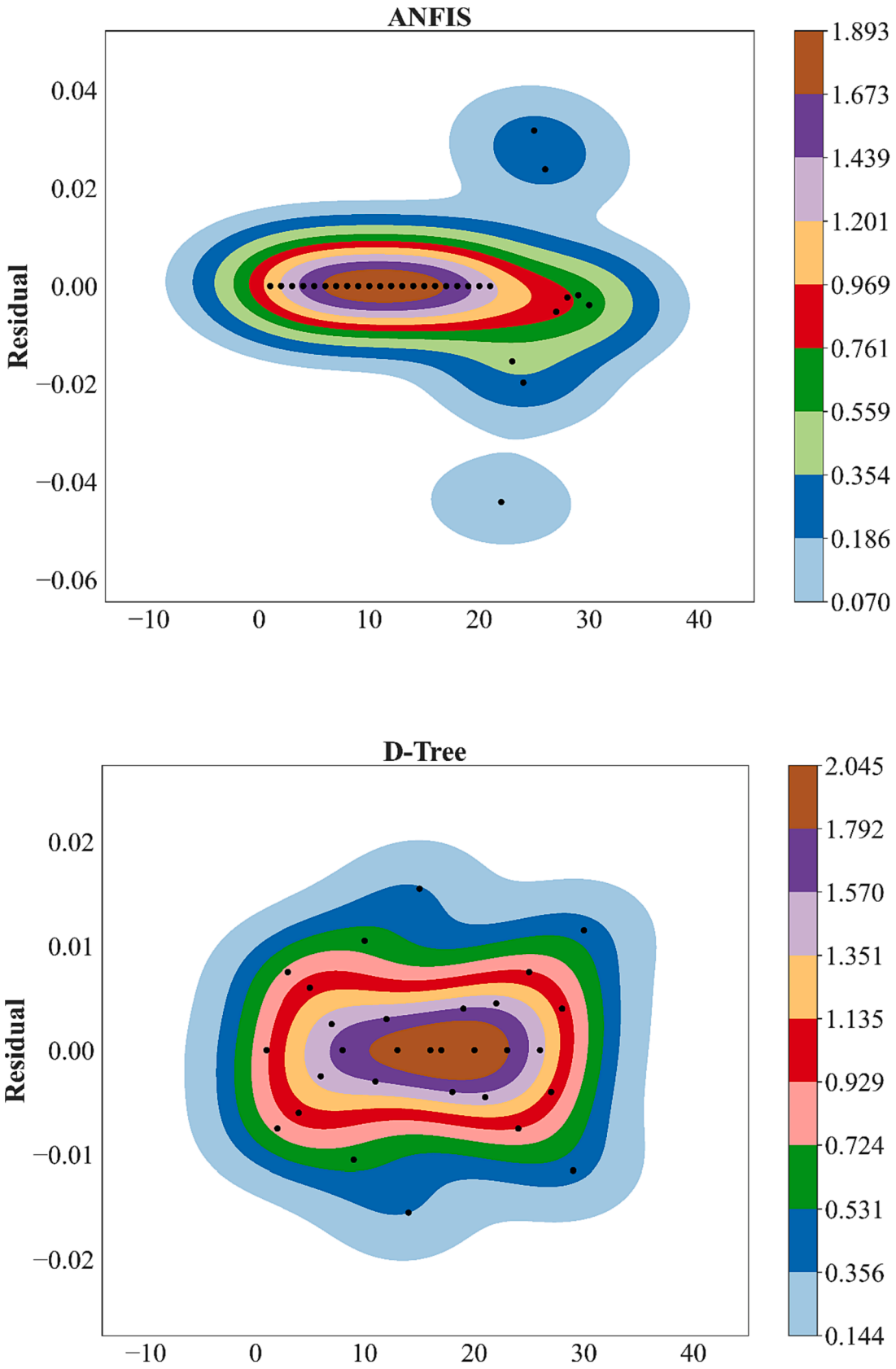


Fig. 4. Residual for TC of MWCNT-CuO-CeO_2 (20-40-40)/water HNF by different algorithms.

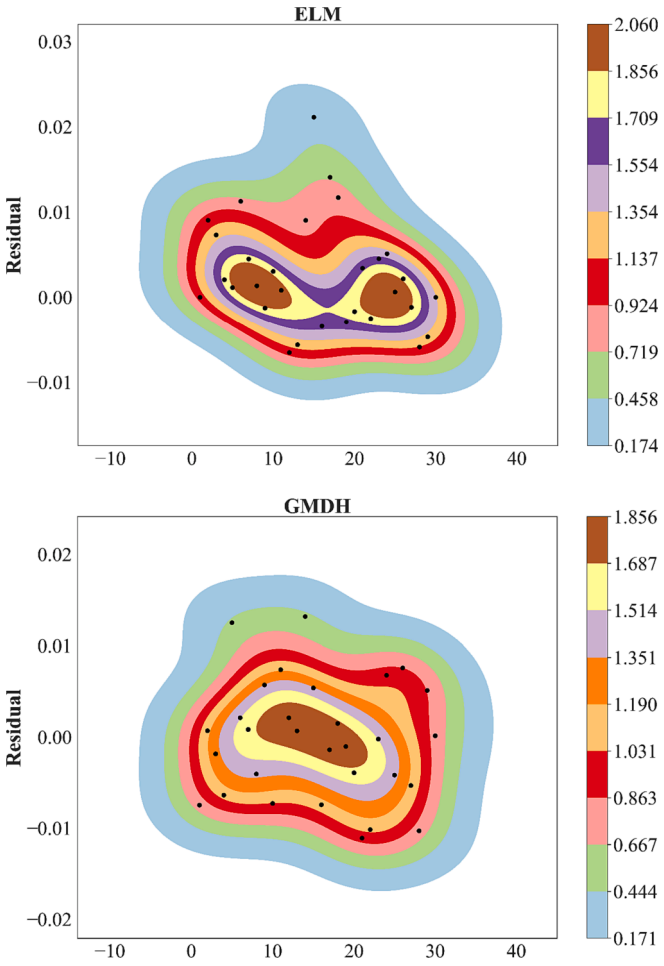


Fig. 4. (*continued*).

$$n_{left} = \left|Q_{left}(\theta)\right| \tag{10}$$

$$n_{right} = \left|Q_{right}(\theta)\right| \tag{11}$$

$$G(Q,\theta) = \frac{n_{left}}{N_m} H\left(Q_{left}(\theta)\right) + \frac{n_{right}}{N_m} H\left(Q_{right}(\theta)\right) \tag{12}$$

The goal is to find the optimal θ to minimize impurities. Similar processing is performed recursively for $Q_{left}(\theta)$ and $Q_{right}(\theta)$. One possible approach to transforming certain nodes into leaves is by establishing a minimum value for N_m, which represents the amount of data contained within a node, or by considering the depth of the tree. By implementing this criterion, if any of these conditions is altered such that the node is halved, the node will not be split and instead will be converted into a leaf. The model usually changes because of these two parameters. At the process starts, the node carries all the data, so Q = D. For regression problems, impurity calculations are performed according to Eqs. (13) and (14).

$$\bar{y}_m = \frac{1}{N_m} \sum_{i \in N_m} y_i \tag{13}$$

$$H(Q) = \frac{1}{N_m} \sum_{i \in N_m} (y_i - \bar{y}_m)^2 \tag{14}$$

2.1.4. Multi-layer Perceptron ANN

MLP ANN is one of the robust ANN algorithms implemented in various technical fields . There are three layers in this algorithm:

- Input layer,
- hidden layer
- output layer

The input layer carries the graph variables and the output layer consists of the goal function. A hidden layer links the input to the output. This is done by neurons, which pass all input data through numerical operations to the output layer. Assuming the input data is represented by the vector $X = (x_1, x_2, x_3, \cdots, x_n)$, all the input data are first represented by the vector $W = (w_1, w_2, w_3, \cdots, w_n)$, a constant value w_0 , known as the bias, is added as can be seen in Eq. (15).

$$c = \sum_{i=1}^{n} x_i w_i + w_0 \tag{15}$$

After the input weighting operation, each c value is predicted by the activation function f, the predicted value is represented by $\hat{y}$, and the difference between this value and the output value of y should be minimal.

$$\hat{y} = f(c) y - \hat{y} \rightarrow min \tag{16}$$

There exists a variety of activation functions that can be utilized for regression procedures. However, among these options, the hyperbolic tangent function (Tansig) and the linear function (Purelin) are the most commonly employed. These functions are mathematically represented by Eqs. (17) and (18), respectively.

$$f(c) = \tanh(c) = \frac{e^{c} - e^{-c}}{e^{c} + e^{-c}} \tag{17}$$

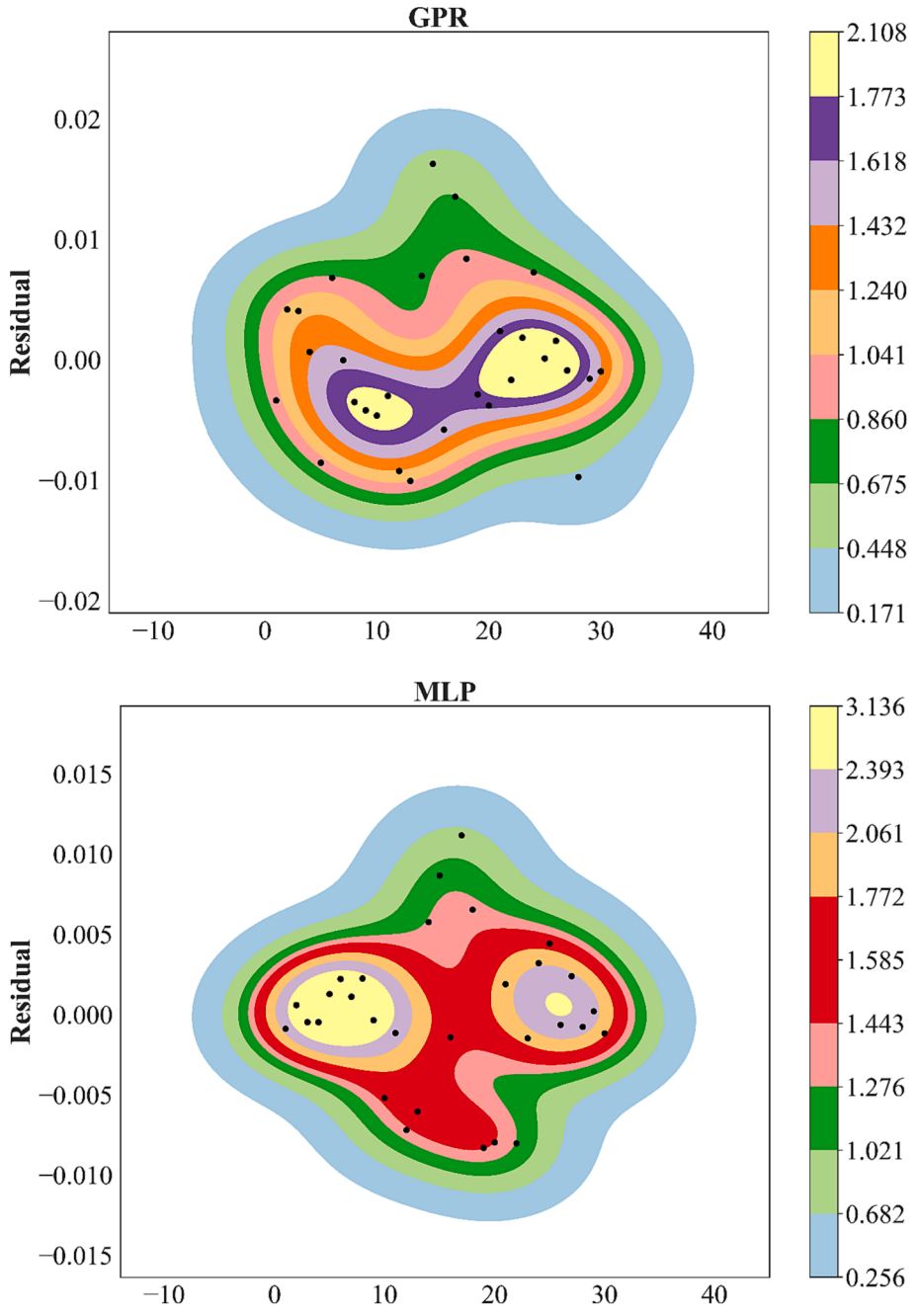


Fig. 4. (*continued*).

$$f(c) = c \tag{18}$$

The process of assigning weights to incoming data and subsequently approximating it using an activation function occurs within the neuron, hence constituting the formation of the first layer within the middle layer of the neuron. The subsequent layer (referred to as the hidden layer) is generated by applying an activation function to the input data, which has been weighted based on predictions made by these neurons. The activation function is applied to the hidden layer of the hyperbolic tangent function as well as the output of the linear function.

2.1.5. *Support vector machine*

SVM is a supervised learning approach utilized for both classification and regression tasks [41–43]. One prevalent application of Support Vector Machines (SVMs) is the organization of data points. Consequently, the plane is regarded as a means of spatially dividing the spots that are farthest away from one another. Subsequently, two planes are deemed to be equidistant from the aforementioned separation plane, serving as markers for demarcating the boundary of the points. This maximizes the distance between these two planes, which are $\frac{2}{\|w\|}$ to the central plane, and also indicates the border of the points. Therefore, $\|\boldsymbol{w}\|$ should be as small as possible. Points lying on these two bounding surfaces are called support vectors. SVMs can be used in both linear and nonlinear methods for classification and regression. The linear SVM of the split plane has a linear equation type, whereas the nonlinear SVM is also a nonlinear vector with Gaussian polynomials. This model attempts to separate the linear equations or planes between points in order to obtain the desired equation and a good regression. To decrease this line, a parameter is described, and for values of this parameter close to zero, the assist vector regression operation is effective. It complicates the regression technique on the one hand. A nonlinear Gaussian functional

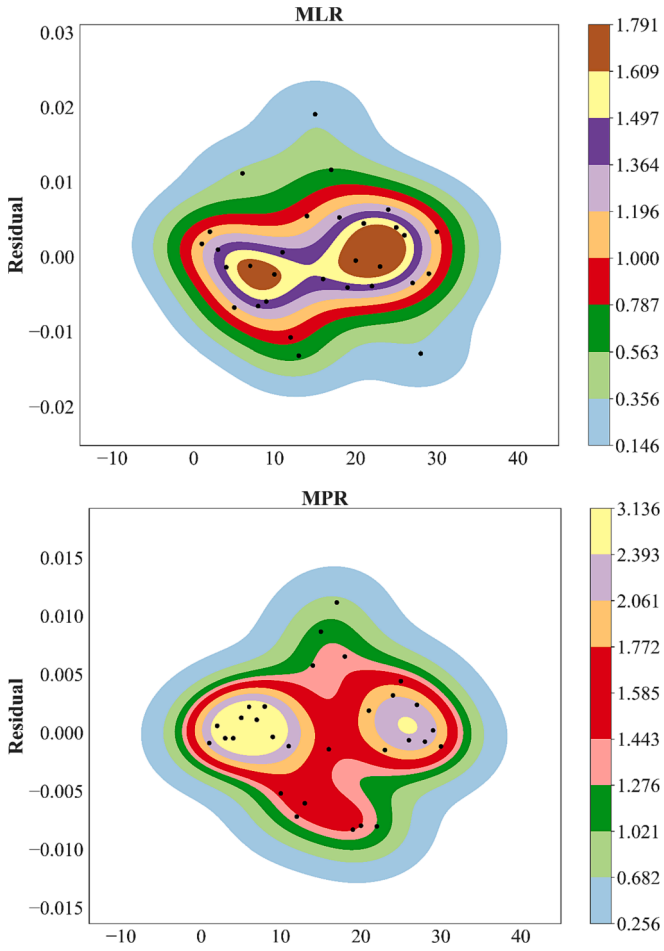


Fig. 4. (*continued*).

regression method is implemented in the current work.

2.1.6. Extreme learning machine

Extreme machine learning is a class of ANN algorithms used for classification, regression, clustering, sparse approximation, compression, and feature learning [44,45]. This algorithm, like other ANN algorithms, contains one or more hidden layers. However, unlike perceptron hierarchical ANNs and radial ANNs, the weights in extreme machine learning ANNs are randomly determined and never updated (programmed randomly but using nonlinear transformations). Use the neurons from the previous layer unchanged. The input matrix is $x = (x_{i1}, x_{i2}, \cdots, x_{i3})$ that $(i = 1, 2, 3, \cdots M)$ and the output is obtained according to Eq. (19).

$$y = f(x_{i1}, x_{i2}, \cdots, x_{i3})(i = 1, 2, 3, \cdots M) \tag{19}$$

ANN predicts values whose function and output are represented by $\widehat{f}$ and $\hat{y}$, respectively. For prediction, the input vector data are first weighted and these weights are randomly selected and denoted as β. This weighted value must then be passed through a mathematical function in order for the predicted value to approach the output value. This function is denoted by h based on Eqs. (20) and (21).

$$h_i(x) = \widehat{f}(a_i, b_i, x) \tag{20}$$

$$\hat{y} = \sum_{i=1}^{N} \beta_i h_i(x) \tag{21}$$

The values denoted as 'a' and 'b' correspond to configurations that are associated with the activator function. In the prediction process, various mathematical or activation functions are utilized, with the tangent hyperbolic function being the most frequently employed. This function is formally stated as Eq. (22).

$$\widehat{f} = \frac{1}{1 + e^{-ax+b}} \tag{22}$$

The matrix H, as defined in equation (23), contains all the output values of the activation function.

$$H = \begin{bmatrix} h(x_1) \\ h(x_2) \\ . \\ . \\ h(x_M) \end{bmatrix} \tag{23}$$

The goal of the ELM algorithm is to estimate H matrix and the weight matrix to predict the output values.

2.1.7. Radial basis function

Radial ANNs is a very effective technique utilized for tasks such as function approximation, prediction, and pattern categorization [45,46]. Moreover, radial ANNs provide significant utility in the modeling of nonlinear data. Similar MLP ANNs, radial ANNs also demonstrate rapid training capabilities. Similar to other ANN techniques, radial ANNs consist of three distinct layers: an input layer, a hidden layer, and an output layer. The input layer is responsible for carrying the input data, while the hidden layer applies the radiation activation function to the data. Ultimately, linear function is executed and the resulting output or objective function is obtained. Within the hidden layer, the input data is gathered and neurons are formed. The aforementioned neurons are

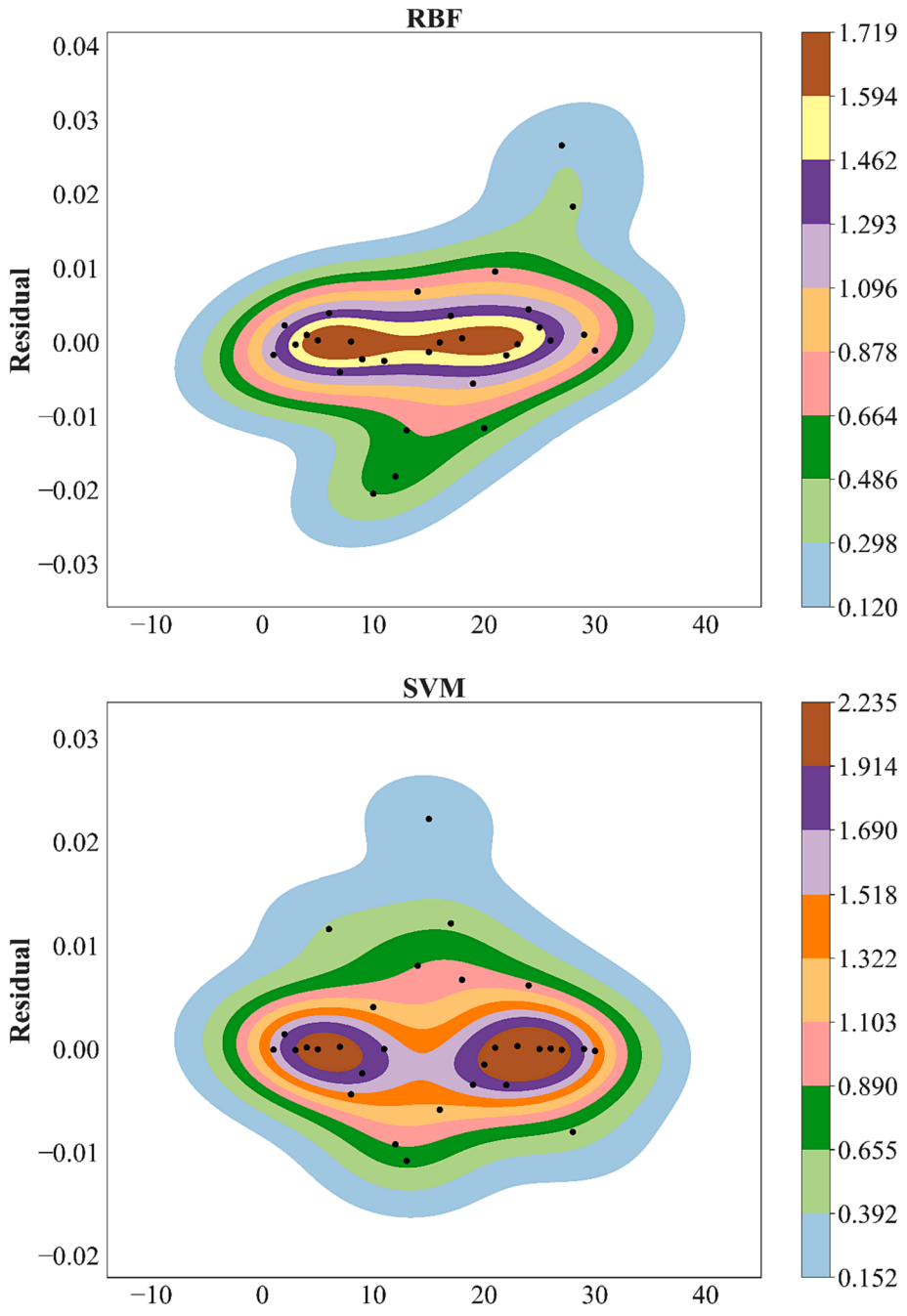


Fig. 4. (*continued*).

characterized by their utilization of radial functions as units for processing their inputs. Subsequently, weights are provided to the perceptrons corresponding to each neuron as inputs to the neurons, similar to the approach employed in a multi-layer ANN. The activation function of the output layer is linear. The RBFs of the ANN are in correspondence with Eq. (24).

$$y_i(\mathrm{x}) = \sum_{j=1}^{n} w_{ij} \varnothing\left(\left\|x - c_j\right\|\right) \tag{24}$$

where x is the input vector, y is the output of the ANN, n is the number of neurons in the hidden layer, c_j is the center of neuron j, and w_{ij} is the weight value of the jth neuron in the hidden layer. The FBR, or Feedback Receptive Field, is implemented within the buried layer neurons. ∅ is a multidimensional radial basis function (RBF) that is capable of demonstrating the distinction between the input vector and the mean vector. There exist multiple radial basis functions (RBFs), with the Gaussian function being the most prevalent among them. It is mathematically stated as Eq. (25).

$$\varnothing\left(\left\|\mathrm{x} - \mathrm{c_j}\right\|\right) = \mathrm{e}^{\left(-\frac{\left\|\mathrm{x}-\mathrm{c_j}\right\|^2}{2\sigma_j^2}\right)} \tag{25}$$

where σ_j is the width of the j_{th} neuron in the hidden layer. The goal of a radial basis function ANN is to find the center widths and weights of the neurons connected from the hidden layer to the output layer so that the value of the objective function or output can be correctly predicted.

2.1.8. Adaptive neuro-fuzzy inference system

ANFIS is an artificial neural network based on the Takagi-Sugeno

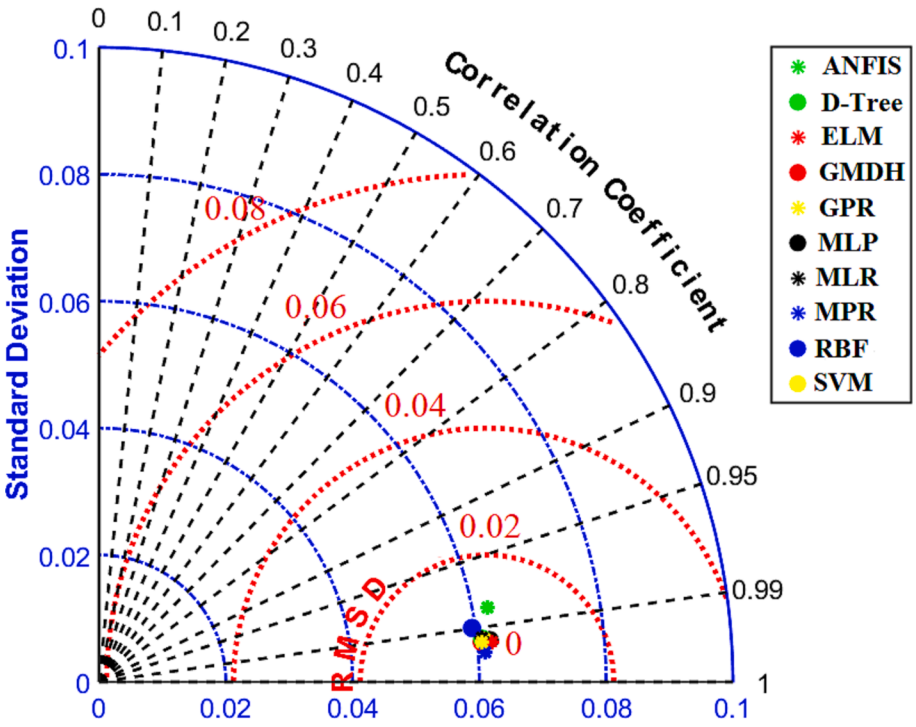


Fig. 5. Data analysis using the Taylor plot.

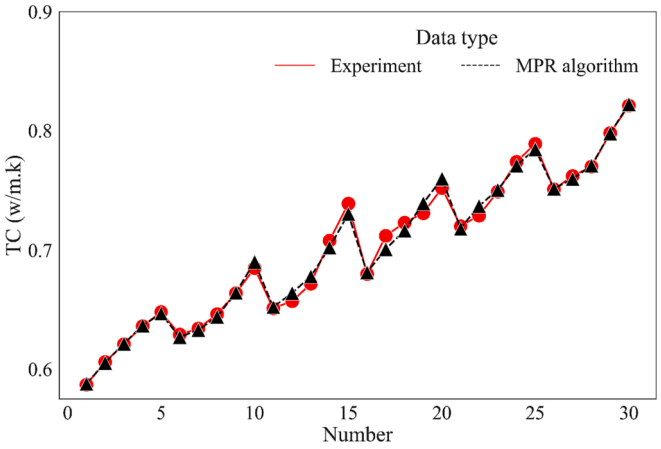


Fig. 6. Comparison of TC predicted by ANN with experimental data.

fuzzy inference system. [45,46]. His logic is equivalent to a set of approximate IF-THEN rules that are designed to approximate nonlinear functions. [47]. Two parts can be identified in the network structure: the antecedent part and the consequent part. To be more explicit, the architecture comprises five distinct layers. The first layer is responsible for receiving input data and subsequently determining the corresponding membership functions. This layer is commonly known as a fuzzification layer. The membership degree for each operation is calculated using the assumed parameter set, denoted as (a, b, c). The second layer is responsible for the production of the magnitude of the rule shots. The subsequent layer is referred to as the "regulatory layer" due to its designated function. The mission of the third layer is to do the normalization of the calculated data. The data is subsequently transmitted to the fourth layer. The received values undergo a phase shift, resulting in a non-phased state. Subsequently, these data are transmitted to the output [48]. The present study employed the ANFIS ANN 10 prediction model, whereby a Gaussian membership function was utilized as the input and a linear membership function was employed as the output.

2.1.9. *Gaussian process regression*

These probabilistic regression models are supported by nonparametric kernels. A Bayesian method generates a probability distribution for all possible values. GPR models are implemented in numerous applications, including variable regression, experimental design, and model fitting, and can also solve nonlinear regression issues [49–52]. The GPR method possesses several advantageous characteristics when compared to other machine learning methods. Firstly, it is straightforward to implement and demonstrates effective performance when dealing with small data sets. Moreover, GPR exhibits self-adaptive capabilities, enabling accurate parameter estimation. Moreover, GPR can be seamlessly integrated with various machine learning tasks, including model training, uncertainty estimation, and hyperparameter estimation. This versatility further enhances the appeal and utility of GPR [53]. Gaussian Process (GP) enables the incorporation of prior information and specifications for the structure of the model through the selection of various kernel functions. The GPR methodology operates under the assumption that the observed value of the dependent variable, denoted as y, may be mathematically represented by Eq. (26).

$$y = f(X(k)) + \varepsilon \tag{26}$$

where the function f represents an unidentified nonlinear operator that is assumed to be well-behaved. The variable ε denotes Gaussian noise, with a variance of $\sigma_n{}^2$,. Lastly, X represents the observed value of the input variable, namely the measured price. Within the realm of function space, there are prior probabilities that can be elucidated by a Gaussian process, wherein the mean is denoted as m(x) and the covariance is represented by $cov(x, x')$ in Eq. (27).

$$f(x)\ GP(m(x), cov(x, x')) \tag{27}$$

In the context of the given variable amount x^*, the response variable can be predicted by utilizing the probability distribution $p(y^*|X, y, x^*)$, which is determined using the mean and variance as described in Eqs.

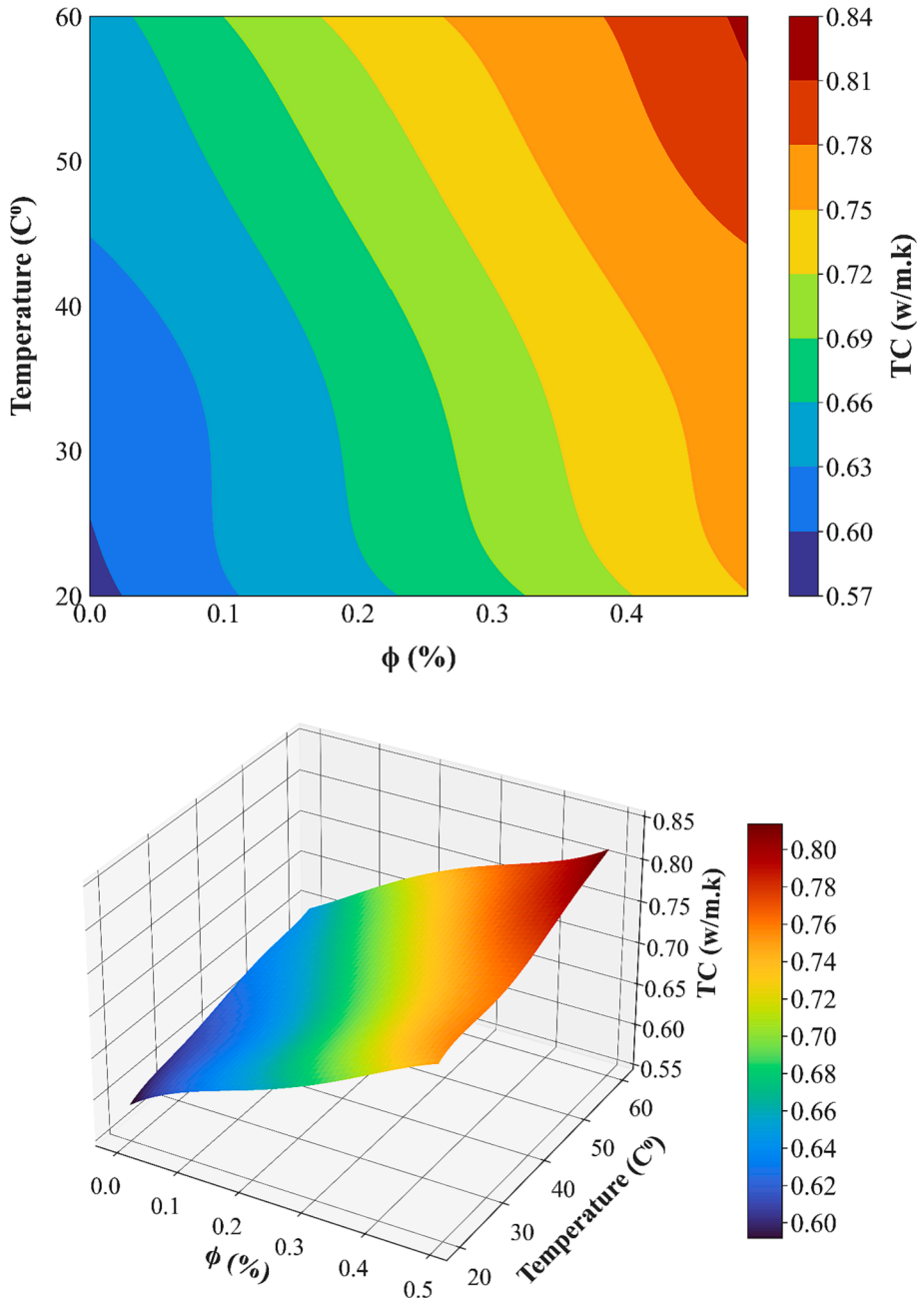


Fig. 7. Surface and contour plot of inputs (φ, T) and TC of MWCNT-CuO-CeO_2 (20-40-40)/water HNF.

(28), 29).

$$\hat{y}^* = m(x^*) + k^{T*}\left(K + {\sigma_n}^2 I\right)^{-1}(y - m(m(x^*))) \tag{28}$$

$${\sigma_{y^*}}^2 = k^* + {\sigma_n}^2 - \left(K + {\sigma_n}^2 I\right)^{-1} k^* \tag{29}$$

where k* represents the vector delineated by $|k^*|_i = cov(x_i, x^*)$, $k^* = cov(x_i, x^*)$ and K represents the variance matrix with parts $|k^*|_{i,j} = cov(x_i, x_j)$ and i represents the scalar matrix. In contrast to traditional regression methods, which rely purely on parameterization to define model predictions, the model output in this case is the training data belonging to dataset X, y. The implementation of these parameters, generally referred to as hyperparameters, entirely determines the predictive probability distributions. Hyperparameter values are obtained by maximizing the log-likelihood operation of the training dataset as delineated in Eq. (30).

$$logp(y|x) = -\frac{1}{2}y^T\left(K + {\sigma_n}^2 I\right)^{-1} y - \frac{1}{2}\log\left(\left|K + {\sigma_n}^2 I\right|\right) - \frac{n}{2}\log(2\pi) \tag{30}$$

2.1.10. Multivariate polynomial regression

Polynomial regression can be employed for a univariate regressor variable, known as simple polynomial regression, or it can be applied to several regressors, known as multiple polynomial regression [54,55]. An example of a second-order multiple polynomial regression can be represented as Equation (31).

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_{11} {x_1}^2 + \beta_{22} {x_2}^2 + \beta_{12} x_1 x_2 + \varepsilon \tag{31}$$

Again, this can be represented in matrix form as following

$$Y = \beta X + \varepsilon \tag{32}$$

The parameters of the given equation can be calculated as Eq. (33):

Table 5
Examining the changes in mutation rate and crossover on response convergence, convergence time, and NFE.

Crossover rate	Mutation rate	NFE	Time (sec)	Best cost	Crossover rate	Mutation rate	NFE	Time (sec)	Best cost
0	0	1100	1.597511	0.81264	0.6	0.1	71,100	6.250931	0.82213
0.1	0	11,100	2.48828	0.81268	0.7	0.1	81,100	7.598963	0.82213
0.2	0	21,100	3.025789	0.80763	0.8	0.1	91,100	4.598289	0.82213
0.3	0	31,100	3.111704	0.80443	0.9	0.1	101,100	5.029778	0.82213
0.4	0	41,100	3.388217	0.81392	1	0.1	111,100	5.13685	0.82213
0.5	0	51,100	3.806781	0.81478	0	0.15	16,100	2.263082	0.82213
0.6	0	61,100	4.348441	0.818	0.1	0.15	26,100	2.744492	0.82213
0.7	0	71,100	4.725215	0.80438	0.2	0.15	36,100	2.970924	0.82213
0.8	0	81,100	4.67458	0.81126	0.3	0.15	46,100	3.245361	0.82213
0.9	0	91,100	4.8116	0.80914	0.4	0.15	56,100	3.910177	0.82213
1	0	101,100	5.467484	0.80813	0.5	0.15	66,100	3.729666	0.82213
0	0.05	6100	1.943656	0.82213	0.6	0.15	76,100	3.828314	0.82213
0.1	0.05	16,100	3.19447	0.82213	0.7	0.15	86,100	4.948961	0.82213
0.2	0.05	26,100	2.913729	0.82213	0.8	0.15	96,100	4.500804	0.82213
0.3	0.05	36,100	4.189357	0.82213	0.9	0.15	106,100	4.956517	0.82213
0.4	0.05	46,100	3.82745	0.82213	1	0.15	116,100	5.542763	0.82213
0.5	0.05	56,100	5.187767	0.82213	0	0.2	21,100	2.875794	0.82213
0.6	0.05	66,100	4.409594	0.82213	0.1	0.2	31,100	3.095718	0.82213
0.7	0.05	76,100	4.581536	0.82213	0.2	0.2	41,100	4.198397	0.82213
0.8	0.05	86,100	4.309941	0.82213	0.3	0.2	51,100	3.759031	0.82213
0.9	0.05	96,100	5.42799	0.82213	0.4	0.2	61,100	4.759423	0.82213
1	0.05	106,100	5.294289	0.82213	0.5	0.2	71,100	4.40728	0.82213
0	0.1	11,100	2.364964	0.82213	0.6	0.2	81,100	4.476238	0.82213
0.1	0.1	21,100	2.73837	0.82213	0.7	0.2	91,100	4.784703	0.82213
0.2	0.1	31,100	5.543064	0.82213	0.8	0.2	101,100	5.10704	0.82213
0.3	0.1	41,100	3.422696	0.82213	0.9	0.2	111,100	5.412035	0.82213
0.4	0.1	51,100	3.650146	0.82213	1	0.2	121,100	5.554066	0.82213
0.5	0.1	61,100	3.723295	0.82213					

Table 6
Optimal point of the GA Algorithm.

φ (%)	T (°C)	TC $\left(\frac{W}{m.K}\right)$
0.5	60	0.82213

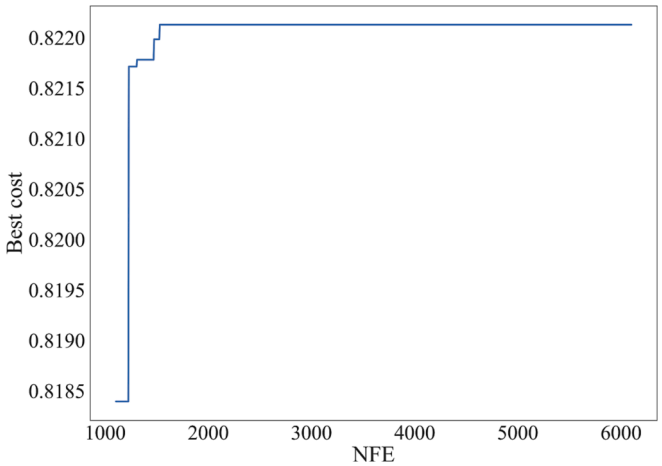


Fig. 8. Convergence flow-chart by genetic algorithm.

$$\beta = \left(\hat{X}X\right)^{-1}\hat{X}Y \tag{33}$$

and the calculated regression equation is expressed as Eq. (34):

$$\hat{y} = \widehat{\beta_0} + \widehat{\beta_1}x_1 + \widehat{\beta_2}x_2 + \widehat{\beta_{11}}{x_1}^2 + \widehat{\beta_{22}}{x_2}^2 + \widehat{\beta_{12}}x_1x_2 \tag{34}$$

2.2. Hyper-parameters and training

Hyperparameters refer to the adjustable parameters within an algorithm that can be optimized in order to enhance the predictive capabilities of the algorithm. For instance, the efficacy of the algorithm can be enhanced by optimizing the configuration of the hidden layer through the adjustment of the number of neurons. Every algorithm possesses distinct hyperparameters that can be adjusted based on its structure. The hyperparameters encompass various factors, such as the quantity of neurons in the hidden layer, the number of hidden layers, the learning rate, the pressure coefficient for neuron selection, and the activation function. In addition to configuring these hyperparameters, it is necessary to segment the data. The data is partitioned into three distinct groups, namely training, validation, and testing. The optimal functioning of all three data categories is crucial, as any error within these categories can significantly compromise the algorithm's effectiveness. This study employs a data allocation scheme where 70% of the available data is designated for training purposes, 15% of the data is allocated for validation, and the remaining portion is reserved for testing.

2.3. Results review criteria

In order to assess the efficacy of a regression algorithm, it is important to analyze its performance against a specific criterion. This article employs several quantitative measures to assess the performance, including the root mean squared error (RMSE), correlation coefficient (R), mean squared error (MSE), R squared (R^2), mean absolute percentage error (MAPE), and mean absolute error (MAE). The aforementioned requirements are delineated in Eqs. (35)–(40).

$$R = \frac{\sum_{i=1}^{n}\left(y_{i,Experiment} - \overline{y}_{i,Experiment}\right)\left(y_{i,ANN} - \overline{y}_{i,ANN}\right)}{\sqrt{\sum_{i=1}^{n}\left(y_{i,Experiment} - \overline{y}_{i,Experiment}\right)^2\sum_{i=1}^{n}\left(y_{i,ANN} - \overline{y}_{i,ANN}\right)^2}} \tag{35}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}\left(y_{i,ANN} - y_{i,Experiment}\right)^2}{n}} \tag{36}$$

$$MAE = \frac{\sum_{i=1}^{n}\left|y_{i,ANN} - y_{i,Experiment}\right|}{n} \tag{37}$$

$$MSE = \frac{\sum_{i=1}^{n}\left(y_{i,ANN} - y_{i,Experiment}\right)^2}{n} \tag{38}$$

$$R^2 = 1 - \sum_{i=1}^{n}\frac{\left(y_{i,ANN} - y_{i,Experiment}\right)^2}{{y_{i,Experiment}}^2} \tag{39}$$

$$MAPE = \frac{\sum_{i=1}^{n}\left|\frac{y_{i,ANN} - y_{i,Experiment}}{y_{i,Experiment}}\right|}{n} * 100 \tag{40}$$

The deviation between the experimental results and the results predicted by the correlation can be calculated as Eq. (41) [56].

$$Marginofdeviation = \left(\frac{\mathrm{TC}_{Exp} - \mathrm{TC}_{ANN}}{\mathrm{TC}_{Exp}}\right) * 100 \tag{41}$$

3. Results

3.1. Data analysis

The study's data is derived from Table 1, with the inputs being φ and temperature. The primary objective is to predict and optimize TC. It is crucial to examine the behavior of data prior to making predictions, as understanding the relationship between the inputs and the output is essential. Fig. 1 illustrates the variations in both the input and output variables. Based on the data presented in Fig. 1, it is evident that there is a positive correlation between the variable φ and the corresponding increase in the value of TC, indicating its utility. Furthermore, it should be noted that temperature exhibits similar characteristics to φ and contributes to the elevation of TC. In order to conduct a more comprehensive examination of the influence of the data, one can employ sensitivity analysis. The Morris sensitivity analysis is widely regarded as a highly effective technique for conducting sensitivity analyses, and it has been extensively employed in several investigations. The objective of this analysis is to demonstrate the impact and involvement of input factors on outcomes. In present study, the variable denoting influence is represented by the symbol μ, whereas the variable representing participation is denoted by the symbol σ. The examination of sensitivity analysis is observable in Table 2. Based on the findings shown in Table 2, it is evident that the variable φ exerted a more substantial influence on TC in comparison to temperature. Furthermore, a method for assessing the impact of input variables on the output is by visualizing the data through a heatmap graph. The diagram depicted in Fig. 2 is observable. Based on the findings presented in Fig. 2 and Table 2, it can be inferred that both input variables exert a substantial influence on alterations in the output.

3.2. Results of machine learning algorithms

To achieve the most effective selection of supervised machine learning algorithms and ANNs, it is recommended to employ optimization techniques such as GA. As previously stated, every algorithm possesses its own set of hyperparameters, which are discussed in Table 3, displaying their respective ranges. The objective of the GA is to minimize the discrepancy between the observed experimental data and the data projected by the predictive algorithm.

Following the selection of the optimal mode for each algorithm through the utilization of a GA, it is now feasible to examine other indicators, as presented in Table 4.

Based on the findings presented in Table 4, it can be observed that the MPR algorithm exhibits the highest level of performance. In order to evaluate the accuracy of this correlation, a bias analysis of the TC ratio is conducted.

Fig. 3 shows the calculated deviation margin for TC of MWCNT-CuO-CeO_2 (20-40-40)/water HNF. It can be found that the maximum margins of deviation related to GPR, MLR, D-Tree, ELM, MPR, MLP, RBF, SVM, ANFIS and GMDH algorithms are equal to 0.031, 0.02579, 0.028946, 0.033889, 0.01568, 0.02515, 0.03485, 0.03, 0.0385 and 0.0178, respectively. Moreover, Fig. 4 shows the KDE plot for the difference between experimental data and the data predicted by regression algorithms. In this diagram, it can be seen that there is less residual dispersion in the MPR algorithm and these residuals are close to zero. The Taylor diagram serves as a supplementary tool for data analysis. The diagram provided is highly commendable for the purpose of doing data analysis. Values that are in close proximity to zero suggest that the data possesses a suitable standard deviation (STD), a correlation coefficient (R) that is close to one, and a root mean square deviation (RMSD) that is close to zero. Hence, when assuming this position, it offers the optimal condition. The Taylor diagram for TC is shown in Fig. 5. According to the findings presented in Table 3, the MPR algorithm demonstrated the highest performance in predicting the attributes of TC, whereas the RBF method exhibited the lowest performance. The values of R, RMSD and STD for MPR and RBF algorithms are 0.9971, 0.0047, 0.061 and 0.9897, 0.0088, 0.0594, respectively. This issue is also evident in the Taylor diagram. According to Table 4 and Figs. 3-5, the MPR algorithm has performed better than other algorithms.

The target of using regressions is to develop a relationship between input and output variables. Eq. (42) shows the relationship between inputs and outputs.

$$\begin{aligned} TC = {} & 0.0812413*T - 0.00437447*T^2 + 0.000115764*T^3 - 1.4791e \\ & - 06*T^4 + 1.0025*SC - 0.00238955*SC*T - 0.00172815*SC*T^2 \\ & + 4.02262e - 05*SC*T^3 - 2.2619e - 07*SC*T^4 - 8.25143*SC^2 \\ & + 0.369602*SC^2*T - 0.00311451*SC^2*T^2 - 8.92857e - 06*SC^2*T^3 \\ & + 18.2139*SC^3 - 0.703234*SC^3*T + 0.00542989*SC^3*T^2 \\ & - 11.1667*SC^4 + 0.277083.*SC^4*T - 0.166667*SC^5 + 7.31019e - 09*T^5 \end{aligned} \tag{42}$$

After obtaining the governing equations between the design variables and the objective functions, the simulation data can be predicted using these equations. By plugging the input values into the equations, the output values for the TC of MWCNT-CuO-CeO_2 (20-40-40)/water HNF can be calculated. The MPR algorithm's predicted values can be compared to experimental data. Fig. 6 illustrates the difference between experimental data and data predicted by the MPR algorithm for the TC of MWCNT-CuO-CeO_2 (20-40-40)/water HNF.

Fig. 7 also displays the surface and contour plots of these objective functions with inputs. It can be observed that the TC of MWCNT-CuO-CeO_2 (20-40-40)/water HNF rises as temperature and φ increase.

3.3. Optimization

The evolutionary algorithm employs generated equations as input [31,42,57–59]. It maximizes the objective function, which is the TC of MWCNT-CuO-CeO_2 (20-40-40)/water HNF. A suitable convergence is accomplished by selecting an initial population and monitoring the changes in mutation rate and crossover rate to test the convergence. In order to examine convergence, an investigation was conducted by altering these two factors, as depicted in Table 5. The convergence of the method is observed when the mutation rate is set to 0.05. However, the convergence time is just prolonged due to the crossover of the mutation rate. This can also be seen in the high rate of intersections. Therefore, the most suitable mutation rate, crossover rate, population size, and some generations for GA algorithm processing are set to be 0.05, 0, 100, and 1000, respectively.

Eq. (43) is used to calculate the range of design variables and objective functions. Furthermore, Table 6 and Fig. 8 show the most optimal point for TC of MWCNT-CuO-CeO_2 (20-40-40)/water HNF.

$$\text{Maximize} \quad TC = f_2(\varphi, T) \tag{43}$$

$$Subjected\ to: \begin{cases} 0 \le \varphi \le 0.5\% \\ 20^{\circ}\mathrm{C} \le T \le 60^{\circ}\mathrm{C} \end{cases}$$

4. Conclusion

The thermal conductivity of MWCNT-CuO-CeO_2 (20-40-40)/water HNF was predicted using a set of 10 regressors in the current study. Both solid volume fraction (φ) and temperature were considered as design variables. The simulation data was predicted using the governing equations established between the design variables and the objective functions. Values for TC of HNF were calculated as a result of plugging the inputs into the appropriate equations. The MPR algorithm's predicted values were compared to actual measurements. According to Morris sensitivity analysis and data analysis graphs, the most important effect was on the output of variable φ and then temperature. Also, the maximum margin of deviation related to GPR, MLR, D-Tree, ELM, MPR, MLP, RBF, SVM, ANFIS and GMDH algorithms was equal to 0.031, 0.02579, 0.028946, 0.033889, 0.01568, 0.02515, 0.03485, 0.03, 0.0385 and 0.0178, respectively. According to the Taylor diagram, MPR was the best algorithm for TC quality prediction and RBF was the worst algorithm. The values of R, RMSD, and STD for the two MPR and RBF algorithms were 0.9971, 0.0047, 0.061 and 0.9897, 0.0088, and 0.0594, respectively. By examining the genetic algorithms parameters such as mutation rate and crossover rate, it was found that the rise in crossover increases the NFE, and this means that more equations are evaluated and the convergence is delayed. Also, up to the mutation rate of 0.05, there were changes in the objective function and it converged afterward. Finally, the best value of TC occurs when the temperature and φ were the highest.

5. Future directions

Viscosity of MWCNT-CuO-CeO_2 (20-40-40)/water hybrid nanofluid (HNF) can be predicted using different regressors in the future studies. Also, shear rate can be considered as a design variable.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] H. Masuda, A. Ebata, K. Teramae, Alteration of thermal conductivity and viscosity of liquid by dispersing ultra-fine particles. Dispersion of Al2O3, SiO2 and TiO2 ultra-fine particles, 1993.

[2] S.U. Choi, J.A. Eastman, Enhancing thermal conductivity of fluids with nanoparticles (No. ANL/MSD/CP-84938; CONF-951135-29). Argonne National Lab.(ANL), Argonne, IL (United States), 1995.

[3] K.V. Sharma, P.H.V.S. Talpa Sai, P. Sharma, P.K. Kanti, P. Bhramara, S. Akilu, Prognostic modeling of polydisperse SiO2/Aqueous glycerol nanofluids' thermophysical profile using an explainable artificial intelligence (XAI) approach, Eng. Appl. Artif. Intel. 126 (106967) (2023), 106967, https://doi.org/10.1016/j.engappai.2023.106967.

[4] P.K. Kanti, A.P. Shrivastav, P. Sharma, M.P. Maiya, Thermal performance enhancement of metal hydride reactor for hydrogen storage with graphene oxide nanofluid: Model prediction with machine learning, Int. J. Hydrogen Energy (2023), https://doi.org/10.1016/j.ijhydene.2023.03.361.

[5] P. Kumar Kanti, P. Sharma, K.V. Sharma, M.P. Maiya, The effect of pH on stability and thermal performance of graphene oxide and copper oxide hybrid nanofluids for heat transfer applications: Application of novel machine learning technique, J. Energy Chem. 82 (2023) 359–374, https://doi.org/10.1016/j.jechem.2023.04.001.

[6] S. Lee, S.S. Choi, S.A. Li, J.A. Eastman, Measuring thermal conductivity of fluids containing oxide nanoparticles, 1999.

[7] H. Xie, M. Fujii, X. Zhang, Effect of interfacial nanolayer on the effective thermal conductivity of nanoparticle-fluid mixture, Int. J. Heat Mass Transf. 48 (14) (2005) 2926–2932.

[8] V. Vărdaru, G. Huminic, A. Huminic, C. Fleacă, F. Dumitrache, I. Morjan, Synthesis, characterization and thermal conductivity of water based graphene oxide–silicon hybrid nanofluids: An experimental approach, Alexandria Eng. J. 61(12) (2022) pp.12111-12122.

[9] R. Bakhtiari, B. Kamkari, M. Afrand, A. Abdollahi, Preparation of stable TiO2-Graphene/Water hybrid nanofluids and development of a new correlation for thermal conductivity, Powder Technol. 385 (2021) 466–477.

[10] M.H. Esfe, E.M. Ardeshiri, D. Toghraie, Experimental study and sensitivity analysis of a new generation of special ternary hybrid nanofluids (THNFs) and investigation of factors affecting its thermal conductivity, Case Stud. Therm. Eng. 34 (2022), 101940.

[11] R. Kumar, P. Kumar, A. Rajan, Thermal performance of automobile radiator under the influence of hybrid nanofluid, Mater. Today:. Proc. 76 (2023) 251–255.

[12] C.H. Li, G.P. Peterson, Mixing effect on the enhancement of the effective thermal conductivity of nanoparticle suspensions (nanofluids), Int. J. Heat Mass Transf. 50 (23–24) (2007) 4668–4677.

[13] R. Karthik, R.H. Nagarajan, B. Raja, P. Damodharan, Thermal conductivity of CuO–DI water nanofluids using 3-ω measurement technique in a suspended micro-wire, Exp. Therm Fluid Sci. 40 (2012) 1–9.

[14] L. Fedele, L. Colla, S. Bobbo, Viscosity and thermal conductivity measurements of water-based nanofluids containing titanium oxide nanoparticles, Int. J. Refrig. 35 (5) (2012) 1359–1366.

[15] J. Gao, J. Liu, H. Yue, Y. Zhao, I. Tlili, A. Karimipour, Effects of various temperature and pressure initial conditions to predict the thermal conductivity and phase alteration duration of water based carbon hybrid nanofluids via MD approach, J. Mol. Liq. 351 (2022), 118654.

[16] H. Adun, D. Kavaz, M. Dagbasi, H. Umar, I. Wole-Osho, An experimental investigation of thermal conductivity and dynamic viscosity of Al2O3-ZnO-Fe3O4 ternary hybrid nanofluid and development of machine learning model, Powder Technol. 394 (2021) 1121–1140.

[17] A. Shahsavar, M. Sepehrnia, H. Maleki, R. Darabi, Thermal conductivity of hydraulic oil-GO/Fe3O4/TiO2 ternary hybrid nanofluid: Experimental study, RSM analysis, and development of optimized GPR model, J. Mol. Liq. (2023), 122338.

[18] H. Guan, Q. Su, R. Wang, L. Huang, C. Shao, Z. Zhu, Why can hybrid nanofluid improve thermal conductivity more? A molecular dynamics simulation, J. Mol. Liq. 372 (2023), 121178.

[19] M.H. Esfe, R. Esmaily, M.K. Khabaz, A.A. Alizadeh, M. Pirmoradian, A. Rahmanian, D. Toghraie, A novel integrated model to improve the dynamic viscosity of MWCNT-Al2O3 (40: 60)/Oil 5W50 hybrid nano-lubricant using artificial neural networks (ANNs), Tribol. Int. 178 (2023), 108086.

[20] Y. Guo, Z. Mustafaoglu, D. Koundal, Spam Detection Using Bidirectional Transformers and Machine Learning Classifier Algorithms, J. Computat. Cognitive Eng. (2022), https://doi.org/10.47852/bonviewJCCE2202192.

[21] H. Gaur, B. Khidhir, R.K. Manchiryal, Solution of structural mechanic's problems by machine learning, Int. J. Hydromechatronics 5 (1) (2022) 22–43.

[22] B.K. Liu, W.Z. Lu, Surrogate models in machine learning for computational stochastic multi-scale modelling in composite materials design, Int. J. Hydromechatronics 5 (4) (2022) 336–365.

[23] S. Rostami, A.A. Nadooshan, A. Raisi, M. Bayareh, Modeling the thermal conductivity ratio of an antifreeze-based hybrid nanofluid containing graphene oxide and copper oxide for using in thermal systems, J. Mater. Res. Technol. 11 (2021) 2294–2304.

[24] X. Yang, A. Boroomandpour, S. Wen, D. Toghraie, F. Soltani, Applying Artificial Neural Networks (ANNs) for prediction of the thermal characteristics of water/ethylene glycol-based mono, binary and ternary nanofluids containing MWCNTs, titania, and zinc oxide, Powder Technol. 388 (2021) 418–424.

[25] A. Pare, S.K. Ghosh, A unique thermal conductivity model (ANN) for nanofluid based on experimental study, Powder Technol. 377 (2021) 429–438.

[26] S.G. Hu, C.H. Li, Z.M. Zhou, B. Liu, Y.B. Zhang, M. Yang, B.K. Li, T. Gao, M.Z. Liu, X. Cui, X.M. Wang, W.H. Xu, Y.S. Dambatta, R.Z. Li, S. Sharma, Nanoparticle -enhanced coolants in machining: mechanism, application and prospects, Frontiers Mechanical Eng. 18 (4) (2023) 53, https://doi.org/10.1007/s11465-023-0769-8.

[27] M. Ramezanizadeh, M. Alhuyi Nazari, Modeling thermal conductivity of Ag/water nanofluid by applying a mathematical correlation and artificial neural network, Int. J. Low-Carbon Technol. 14 (4) (2019) 468–474.

[28] PVR, N.K., S. Venkatachalapathy, P. Kalidoss, P. Chaupal, Experimental Investigation with ANN Modeling of Thermal Conductivity and Viscosity of a Ternary Nanofluid at Different Mixing Ratios and Volume Concentrations, J. Mol. Liquids 383 (2023) p.122006.

[29] S. Rostami, R. Kalbasi, N. Sina, A.S. Goldanlou, Forecasting the thermal conductivity of a nanofluid using artificial neural networks, J. Therm. Anal. Calorim. 145 (2021) 2095–2104.

[30] A.B. Colak, Experimental study for thermal conductivity of water-based zirconium oxide nanofluid: developing optimal artificial neural network and proposing new correlation, Int. J. Energy Res. 45 (2) (2021) 2912–2930.

[31] B. Peng, et al., 3D-STCNN: Spatiotemporal Convolutional Neural Network based on EEG 3D features for detecting driving fatigue, J. Data Sci. Intell. Sys. (2023), https://doi.org/10.47852/bonviewJDSIS3202983.

[32] B. Yang, et al., Enhancing direct-path relative transfer function using deep neural network for robust sound source localization, CAAI Trans. Intell. Technol. 7 (3) (2022) 446–454, https://doi.org/10.1049/cit2.12024.

[33] B. Fan, et al., Intelligent vehicle lateral control based on radial basis function neural network sliding mode controller, CAAI Trans. Intell. Technol. 7 (3) (2022) 455–468, https://doi.org/10.1049/cit2.12075.

[34] L. Ma, et al., Apple grading method based on neural network with ordered partitions and evidential ensemble learning, CAAI Trans. Intell. Technol. 7 (4) (2022) 561–569, https://doi.org/10.1049/cit2.12140.

[35] P. Preethi, H.R. Mamatha, Region-based convolutional neural network for segmenting text in epigraphical images, Artificial Intell. Applic. 1 (2) (2023) 119–127, https://doi.org/10.47852/bonviewAIA2202293.
[36] N. Luo, et al., Fuzzy logic and neural network-based risk assessment model for import and export enterprises: A review, J. Data Sci. Intell. Sys. (2023), https://doi.org/10.47852/bonviewJDSIS32021078.
[37] Z. Chen, Research on Internet Security Situation Awareness Prediction Technology based on Improved RBF Neural Network Algorithm, J. Computational Cognitive Eng. (2022), https://doi.org/10.47852/bonviewJCCE149145205514.
[38] A. Alizadeh, et al., Evaluation of the effects of the presence of ZnO -TiO2 (50 %–50 %) on the thermal conductivity of Ethylene Glycol base fluid and its estimation using Artificial Neural Network for industrial and commercial applications, J. Saudi Chemical Soc. 27 (2) (2023) 101613.
[39] X. Dai, et al., Using Gaussian Process Regression (GPR) models with the Matérn covariance function to predict the dynamic viscosity and torque of SiO2/Ethylene glycol nanofluid: A machine learning approach, Eng. Applic. Artificial Intell. 122 (106107) (2023) 106107.
[40] M. Sabah, M. Talebkeikhah, F. Agin, F. Talebkeikhah, E. Hasheminasab, Application of decision tree, artificial neural networks, and adaptive neuro-fuzzy inference system on predicting lost circulation: A case study from Marun oil field, J. Pet. Sci. Eng. 177 (2019) 236–249.
[41] M. Sharifzadeh, A. Sikinioti-Lock, N. Shah, Machine-learning methods for integrated renewable power generation: A comparative study of artificial neural networks, support vector regression, and Gaussian Process Regression, Renew. Sustain. Energy Rev. 108 (2019) 513–538.
[42] D.J.S. Raj, J.V. Ananthi, Recurrent neural networks and nonlinear prediction in support vector machines, J. Soft Comput. Paradigm 1 (1) (2019) 33–40.
[43] M. Rostamzadeh-Renani, M. Baghoolizadeh, R. Rostamzadeh-Renani, D. Toghraie, B. Ahmadi, The effect of canard's optimum geometric design on wake control behind the car using Artificial Neural Network and Genetic Algorithm, ISA Trans. 131 (2022) 427–443.
[44] S.S. Roy, R. Roy, V.E. Balas, Estimating heating load in buildings using multivariate adaptive regression splines, extreme learning machine, a hybrid model of MARS and ELM, Renew. Sustain. Energy Rev. 82 (2018) 4256–4268.
[45] W.J. Niu, Z.K. Feng, B.F. Feng, Y.W. Min, C.T. Cheng, J.Z. Zhou, Comparison of multiple linear regression, artificial neural network, extreme learning machine, and support vector machine in deriving operation rule of hydropower reservoir, Water 11 (1) (2019) 88.
[46] R.R. Kouser, T. Manikandan, V.V. Kumar, Heart disease prediction system using artificial neural network, radial basis function and case based reasoning, J. Comput. Theor. Nanosci. 15 (9–10) (2018) 2810–2817.
[47] H. Azimi, H. Bonakdari, I. Ebtehaj, Design of radial basis function-based support vector regression in predicting the discharge coefficient of a side weir in a trapezoidal channel, Appl. Water Sci. 9 (2019) 1–12.
[48] D. Karaboga, E. Kaya, Adaptive network based fuzzy inference system (ANFIS) training approaches: a comprehensive survey, Artif. Intell. Rev. 52 (2019) 2263–2293.
[49] A. Girard, C. Rasmussen, J.Q. Candela, R. Murray-Smith, Gaussian process priors with uncertain inputs application to multiple-step ahead time series forecasting, Adv. Neural Inform. Process. Syst. 15 (2002).
[50] C.E. Rasmussen, Gaussian processes in machine learning, in: Summer School on Machine Learning, Springer, Berlin Heidelberg, Berlin, Heidelberg, 2003, pp. 63–71.
[51] S. Chen, K. Mihara, J. Wen, Time series prediction of CO2, TVOC and HCHO based on machine learning at different sampling points, Build. Environ. 146 (2018) 238–246.
[52] D. Fang, X. Zhang, Q. Yu, T.C. Jin, L. Tian, A novel method for carbon dioxide emission forecasting based on improved Gaussian processes regression, J. Clean. Prod. 173 (2018) 143–150.
[53] J. Hu, J. Wang, Short-term wind speed prediction using empirical wavelet transform and Gaussian process regression, Energy 93 (2015) 1456–1466.
[54] H. Gatignon, Statistical analysis of management data, Kluwer Academic Publishers, Boston, MA, 2003.
[55] D.G. Kleinbaum, L.L. Kupper, A. Nizam, E.S. Rosenberg, Applied regression analysis and other multivariable methods, Cengage Learning, 2013.
[56] M.H. Esfe, S. Saedodin, An experimental investigation and new correlation of viscosity of ZnO–EG nanofluid at various temperatures and different solid volume fractions, Exp. Therm Fluid Sci. 55 (2014) 1–5.
[57] P.K. Kanti, P. Sharma, B. Koneru, P. Banerjee, K.D. Jayan, Thermophysical profile of graphene oxide and MXene hybrid nanofluids for sustainable energy applications: Model prediction with a Bayesian optimized neural network with K-cross fold validation, FlatChem 39 (100501) (2023), 100501, https://doi.org/10.1016/j.flatc.2023.100501.
[58] M. Baghoolizadeh, M. Rostamzadeh-Renani, S.A.H.H. Dehkordi, R. Rostamzadeh-Renani, D. Toghraie, A prediction model for CO2 concentration and multi-objective optimization of CO2 concentration and annual electricity consumption cost in residential buildings using ANN and GA, J. Clean. Prod. 379 (2022), 134753.
[59] M. Baghoolizadeh, M. Rostamzadeh-Renani, R. Rostamzadeh-Renani, D. Toghraie, Multi-objective optimization of Venetian blinds in office buildings to reduce electricity consumption and improve visual and thermal comfort by NSGA-II, Energ. Build. 278 (2023), 112639.