



# Estimation of thermophysical property of hybrid nanofluids for solar Thermal applications: Implementation of novel Optimizable Gaussian Process regression (O-GPR) approach for Viscosity prediction

Humphrey Adun<sup>1</sup> · Ifeoluwa Wole-Osho<sup>1</sup> · Eric C. Okonkwo<sup>2</sup> · Tonderai Ruwa<sup>1</sup> · Terfa Agwa<sup>3</sup> · Kenechi Onochie<sup>3</sup> · Henry Ukwu<sup>4</sup> · Olusola Bamisile<sup>5</sup> · Mustafa Dagbasi<sup>1</sup>

Received: 15 September 2021 / Accepted: 30 January 2022 / Published online: 11 March 2022  
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2022

## Abstract

Solar energy technologies represent a viable alternative to fossil fuels for meeting increasing global energy demands. However, to increase the production of solar technologies in the global energy mix, the cost of production should be as competitive as other sources. This study focuses on the implementation of machine learning for estimating the thermophysical properties of nanofluids for nanofluid-based solar energy technologies as this would make the synthesis of nanofluids cost-effective. The prediction of thermal conductivity has gained a lot of research attention, whereas, the viscosity of nanofluids has less concentration of studies. The accurate prediction of the viscosity of hybrid nanofluids is important in estimating the heat transfer performance of nanofluids as regards their pump power requirements and convective heat transfer coefficient in several applications. The rigor of experimentations of hybrid nanofluids has necessitated the need for developing efficient and robust machine learning models for accurately estimating the viscosity of hybrid nanofluids for solar applications. Several studies were aimed at developing a predictive model for the viscosity of nanofluids; however, these models are limited to specific types of nanofluids. This study is aimed at developing a robust machine learning algorithm for predicting the viscosity of several hybrid nanofluids from reliable experimental data (700 datasets) culled from literature. This study implements a novel optimizable Gaussian process regression (O-GPR), which have not been previously used in this area, and compares the result with other commonly used machine learning algorithms like, Boosted tree regression (BTR), Artificial neural network (ANN), support vector regression (SVR), to accurately predict the viscosity of a wide range of Newtonian-based hybrid nanofluid. The input parameters used in training the machine learning models were temperature (T), volume fraction (VF), the acentric factor of the base fluid (ACF), nanoparticle size (NS), and nanoparticle density (ND). The prediction performance of the machine learning algorithms was tested using statistical metrics and was compared with theoretical models. The O-GPR model showed superior predictive performance with an  $R^2$  of 0.999998 and an  $MSE$  of 0.0002552. The study conclusively states that the high accuracy prediction of thermophysical properties of nanofluid using robust machine learning models makes the design of nanofluid-based solar energy technologies more cost-effective.

**Keywords** Viscosity · Hybrid nanofluid · Artificial neural network · Support vector regression · Nanoparticles

## 1 Introduction

The science of nanoparticle dispersions into base fluids has the potentials to revolutionize various engineering applications. This is due to the difference in the thermophysical property behavior of nanofluids as compared to conventional fluids. In recent times, the thermal conductivity and

viscosity of nanofluids have been of interest to researchers in the field of nanoscience. The viscosity of nanofluids affects both the thermal and lubricative applications of nanofluids. The practical use of nanofluids in thermal management systems is focused on a trade-off between their high thermal conductivity and low viscosity, which is influenced by nanoparticle loadings, size and shape, fluid form, and temperature [1].

Extended author information available on the last page of the article

The viscosity of nanofluids is known to have a significant impact on solar energy applications [2], as it directly impacts the pressure drop and pump work of the system [3]. The accurate knowledge of the viscosity of nanofluids informs better decisions for their applications in solar energy technologies. A study by Asadi et al. [4] reported that for solar applications, the nanofluid studied was not reasonable for applications at a temperature of 55 °C, and a volume concentration of 0.125%, as it resulted in the highest pressure drop and pumping power. In another study by Sarafraz et al. [5], it was shown that despite the improvement in the efficiency of the PV/T system, there was a net negative of pressure drop and pumping power at a volume concentration of 0.3wt%. The viscosity parameter is also used in estimating the Reynold number, which is useful for heat and flow characteristics [6], as shown in Eq. 1. Furthermore, viscosity affects heat transfer enhancement of nanofluids from forced convection and natural convection and occurs in several dimensionless numbers and coefficients such as Reynolds number, Rayleigh number, Prandtl number, Brinkman number, and Colburn j factor used in thermal and fluids sciences [7].

The deviations observed between predictive models and experiments are likely because the models proposed do not account for all known factors that affect the viscosity of nanofluids. These factors include the base fluid viscosity, nanoparticle shape, rate of aggregation, nanoparticle volume concentration, particle mixture ratio, etc. Each of these factors exhibits different relationships with the viscosity of nanofluids. For example, many studies have reported that the viscosity of nanofluids decreases with the rise of fluid temperature [7–14]. In recent years, the attention of researchers has shifted from conventional nanofluids to hybrid nanofluids due to their superior thermophysical properties [15]. In a study by Giwa et al. [16], experimental analysis was done to investigate the effects of base fluid, temperature, and concentration on Al<sub>2</sub>O<sub>3</sub>–Fe<sub>2</sub>O<sub>3</sub> hybrid nanofluid. Their study showed that the viscosity of the hybrid nanofluid was enhanced by 3.23–43.64%. The effects of particle concentration and temperature on hybrid ZnO–MWCNTs were investigated by Marjan et al. [17]. Their study also agreed with the consensus that the viscosity of nanofluids increases with an increase in particle loading, and decreases with an increase in temperature.

Despite the accuracy of these experimental studies in analyzing the viscosity of hybrid nanofluids and the effects of other different parameters on them, these experiments pose the challenge of time and cost. Many of the solution proffered by researchers in predicting the viscosity of nanofluids has been based on theoretical computations and soft-computing methods. A theoretical model for estimating the viscosity of nanofluids with low volume fraction was propounded by Einstein [18]. Other studies [19, 20]

have presented traditional correlation models for estimating the viscosity of nanofluids. These theoretical models have been proven to underestimate the viscosity of nanofluids due to the models not incorporating several parameters that affect the rheological behaviors of the nanofluids [21]. In the recent past, machine learning and data mining tools have been used extensively to predict the relative viscosity of different hybrid nanofluids under varying experimental conditions. The machine learning tools that have been used to predict the relative viscosity of nanofluids includes artificial neural network (ANN) [22], ANFIS-GA [23], and support vector regression (SVR) [24, 25]. More recently, general machine learning models have been developed for predicting nanofluid viscosity. These models were developed using the data mining process from a wide range of different experimental studies on nanofluid synthesis. A study by Alrashed et al. [26] developed using ANFIS, and ANN models for estimating the viscosity of carbon-based nanofluids. Their study used 129 experimental data for optimal prediction using the ANN model. Similarly, a study by Mehrdad et al. [27] developed ANN model with 24 different ANN architectures for predicting hybrid non-Newtonian nanofluids of iron and copper in a base fluid mixture of water and ethylene glycol. Their study showed that the Bayesian regularization (BR) method gave a better performance of viscosity estimation. Furthermore, their study concluded that increasing the hidden neurons of the ANN model slightly increased the model performance. A study by Hossein et al. [28] compared several machine learning models for estimating the dynamic viscosity of CuO/water nanofluid. Their study which developed machine learning methods like MPR, MARS, ANN-MLP, GMDH, and M5-tree based on input parameters of showed that temperature, concentration, and size of nanostructures, showed that ANN-MLP gave the optimum prediction accuracy. A study by Amin et al. [17] used a novel approach of GMDH type of artificial neural network in predicting the viscosity of Fe<sub>3</sub>O<sub>4</sub> nanoparticles. Their result showed an RMSE value of 0.0018 [29–43].

So far, there have been several research articles that apply different intelligent methods for predicting the viscosity of nanofluids. However, previous papers focus mainly on specific hybrid nanofluids. Also, several studies on viscosity prediction do not utilize the nanofluid density as a parameter, which affects the accuracy of the model [44]. The objective of this study is to develop a robust model capable of predicting the relative dynamic viscosity of multiple Newtonian hybrid nanofluids synthesized using different base fluids. To achieve this, over five hundred data points were collected from different experimental studies. These data points were used to train and test different intelligent prediction algorithms. The primary input

parameters used were temperature, volume fraction, mixture ratio, particle size, and the acentric factor of the base fluid, which are known to affect the viscosity of hybrid nanofluids for solar energy applications. It is important to emphasize that the mixture ratio has been neglected by previous researchers when developing machine learning algorithms for estimating the relative viscosity of nanofluids. Mixture ratio is the ratio of individual nanoparticle concentration in the total mix of the hybrid nanofluid. Experimental studies [45–47] have shown that to retrieve optimum thermophysical behaviors of hybrid nanofluids, with different nanoparticle types, an ideal mixture ratio should be known. The effect of mixture ratio on thermophysical properties of hybrid nanofluids was first explained by Hamid et al. [45]. In their study, it was shown that the 1:1 mixture ratio of TiO<sub>2</sub>-SiO<sub>2</sub> hybrid nanofluid gave the highest relative viscosity. A contrary viscosity behavior was shown in a study by Wole-Osho et al. [21]. In their study, the least viscosity was measured for the 1:1 Al<sub>2</sub>O<sub>3</sub>-ZnO mixture ratio, while the 2:1 gave the maximum viscosity. The varying results of the mixture ratio effects on viscosity show that it is a significant factor for their accurate estimation.

The machine learning algorithms used in this study are Optimizable Gaussian process regression (O-GPR), Boosted tree regression (BTR), Artificial Neural Network (ANN) [48], and Support vector regression (SVR), and their performance is compared with other classical correlations using statistical indices. The choice of the machine learning algorithms is based on the accurate estimation of thermophysical properties in the literature. A study by Mehdi et al. [49] reported accurate results with Gaussian process regression for estimating the specific heat capacity of nanofluids. Similarly, ANN and SVR models have been shown to acutely estimate the thermal conductivity of nanofluids [50]. Similarly, a study by Bernani et al. [51] showed that boosted tree regression (BTR) gave an accurate prediction of the heat capacity of nanofluids. The general framework of the prediction is shown in Fig. 1.

It is of worthy note that the utilization of machine learning algorithms has also been developed in the literature to explain the behavior of nanofluids for heat transfer purposes. A study by Huawei et al. [52] developed an optimizable ANN model with multi-objective optimization of GA Pareto optimal front, for estimating the behavior of a non-Newtonian nanofluid composed of Fe<sub>3</sub>O<sub>4</sub> nanoparticles dispersed in liquid paraffin. Their study optimized the viscosity of the nanofluid for maximizing the heat transfer coefficient and minimizing the pressure drop. Their study also concluded that the type of base fluid used influences the thermal and hydrodynamic properties of the thermophysical property like the viscosity. This study concentrates on the thermal utilization of the proposed machine

learning model because the emphasis on the accurate experimental procedure for hybrid nanofluid synthesis can be resolved using improved computational tools, rather than classical models.

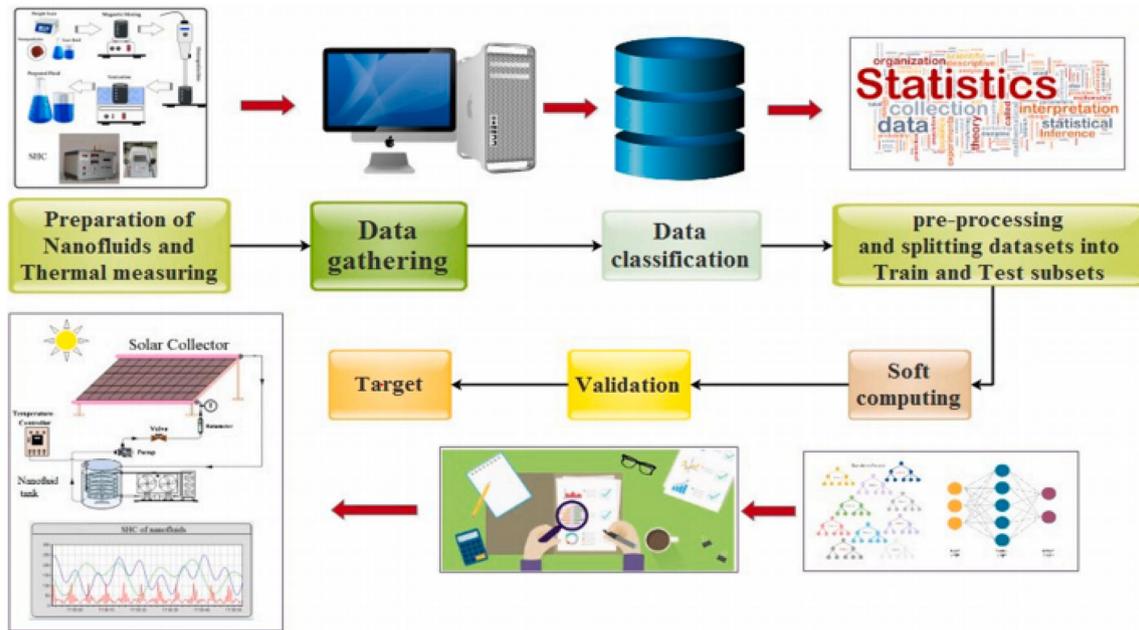
## 2 Methodology

In this study, an intelligent approach to predict the relative viscosity (also referred to as dynamic viscosity ratio,  $\left(\frac{\mu_{nf}}{\mu_{bf}}\right)$ ) of different hybrid nanofluids is presented. The relative viscosity is a dimensionless quantity to express the ratio of the viscosity of a solution (nanofluid) containing a solute to the viscosity of the pure solvent (base fluids). The use of machine learning and deep learning algorithm for this prediction task is considered. In this section, the development of the Optimizable Gaussian Process regression (O-GPR), Boosted tree regression (BTR), ANN, and SVR models are justified and the parameters used for the model implementation are presented. First, the models are explained briefly, before the discourse of how they are applied in predicting the viscosity of hybrid nanofluids is made.

### 2.1 Artificial neural network (ANN)

In recent years, the use of ANN models for nonlinear system prediction has gained more research validation due to some advantages like its low cost, high precision, and speed over other mathematical models. ANN was developed to mirror the performance of the human neural networks (brain) [53]. Typically, ANN algorithms exist as an organized layer that is comprised of interconnected input nodes, output nodes, and hidden layers [54]. It has been widely applied for various purposes such as control systems [55], pattern recognition/data processing [12], and image processing [56]. It has also been used for intelligent prediction of the thermal and chemical properties of different novel hybrid nanofluids [57–61]. While the model architectures show the connection between the neurons and the layers, the weight and biases on these connections can only be determined by the learning algorithm [62]. Figure 2 highlights the general process flowchart for the ANN models.

Specifically, the ANN models used for hybrid nanofluid viscosity prediction in this study consist of one or two hidden layers. The input parameters also known as input layers are listed in Table 1 and include temperature (°C), volume fraction, particle size (nm), mixture ratio, the acentric factor of based fluid, and density (g/m<sup>3</sup>). The number of training, testing, and validation data set is also shown in Table 1. These parameters are used to train the



**Fig. 1** Framework of viscosity prediction for solar energy application using data-driven models

model to predict the relative viscosity (which is the output layer) for different hybrid nanofluids as shown in Fig. 3. A feed-forward backpropagation multilayer perceptron has been developed in this study [62]. Although the Levenberg–Marquardt (LM) training algorithm has been the most preferable algorithm in recent literature [62–64], four different learning algorithms are compared in this study to determine the most suitable model. These include Levenberg–Marquardt (LM), Bayesian Regularization (BR), Gradient Descent with Momentum (GDM), and Scaled Conjugate Gradient (SCG). The mathematical representation of the ANN model is presented in Eq. (1).

$$y(x) = L\left(\sum_{j=1}^N w_j(p) \cdot x_j(p) + c\right) \quad (1)$$

where  $L(\cdot)$  is the hidden transfer function,  $y(x)$  represents the predicted relative viscosity,  $w_j(p)$  is the connection of the neurons in the input layer,  $c$  represents the neuronal bias, and  $x_j(p)$  is the input variable. In this study, the transfer function of the ANN hidden layer is considered as a tangent sigmoid as presented in Eq. (2).

$$f(x) = \frac{1}{1 + \exp(-x)} \quad (2)$$

## 2.2 Support vector regression (SVR)

SVR is a supervised machine learning algorithm that was developed based on statistical learning theory [65]. Unlike linear or simple regression where the aim is to minimize

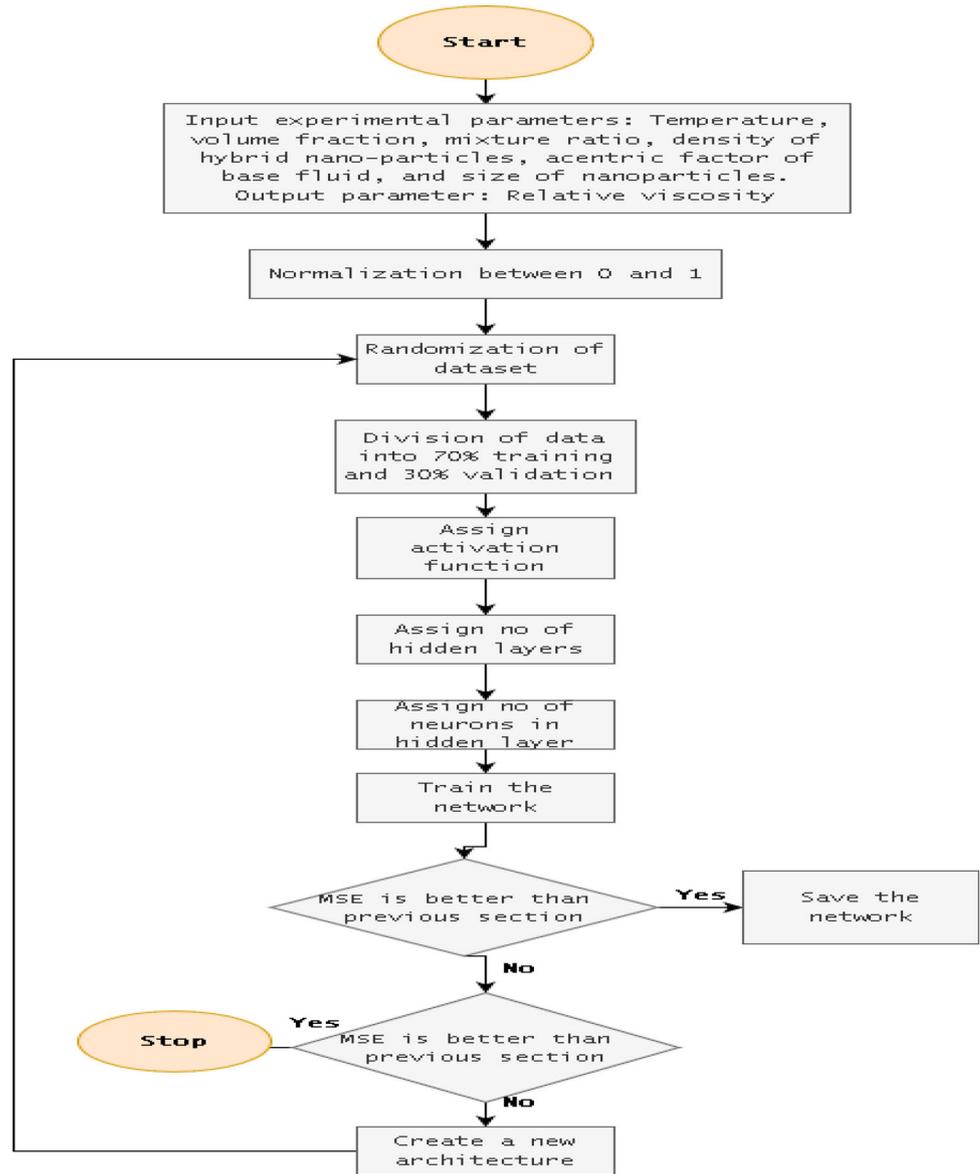
error rate, SVR is used to fit error within a certain threshold. It is developed on the elements of SVM, whereby the data points about a hyperplane are distinctly segregated with support vectors that are the closest points to the generated hyperplane in  $n$ -dimensional feature space [66]. This model is also a popular choice for curve fitting and prediction of linear/nonlinear regression types [67]. An outstanding feature of SVR is its applicability to different disciplines and its convergence speed. In existing works of literature, SVR has been used in engineering science [76], biomedical [77], and social science [78]-related researches. Specifically, it has been used for Covid-19 cases prediction [66], CO<sub>2</sub> sequestration study [68], solar irradiance prediction [79], hybrid nanofluid conductivity prediction [32], and other regression tasks. SVR models differ based on the regularization term used for the structural complexity and the specific choice of loss function used in measuring the empirical risk [80]. In this study, the developed SVR is used to predict the relative viscosity of hybrid nanofluids and the general equation for this model is presented in Eq. (3), where  $b$  is the intercept at  $X = 0$ , and  $w$  is the weight [66].

$$y = wX + b \quad (3)$$

## 2.3 Optimized Gaussian process regression

The process of formulating probabilistic regression from a training data set of  $D = \{x_i, y_i | i = 1, \dots, n\}$  of  $n$  pairs of vectorial inputs  $x_i$  and noisy outputs  $y_i$ , involves computation of the predictive distribution of the functional values

**Fig. 2** ANN processing flowchart



**Table 1** Parameter ranges of the variables and data sets used in the prediction model development [48, 50, 74]

Parameters	Data
Temperature (°C)	5–80
Volume concentration	0–3
Particle size (nm)	4–40 nm
Mixture ratio	0.1–0.9
Acentric factor of base fluid	0.343–0.714
Nanoparticle density (g/cm <sup>3</sup> )	0.25–10.5
Nanofluid relative viscosity	0.815–2.676
Training and validation data	70% (490)
Testing data	30% (210)
Total data	100

$f_*$  or the noisy  $y_*$  at the test location  $x_*$ . Assuming that the noise is additive, independent, or Gaussian, such that the latent function  $f(x)$  and observed noisy targets  $y$  are related, the relationship is expressed as [69]

$$y_i = f(x_i) + \varepsilon_i, \text{ where } \varepsilon_i \sim \mathcal{N}(0, \sigma_{noise}^2) \tag{4}$$

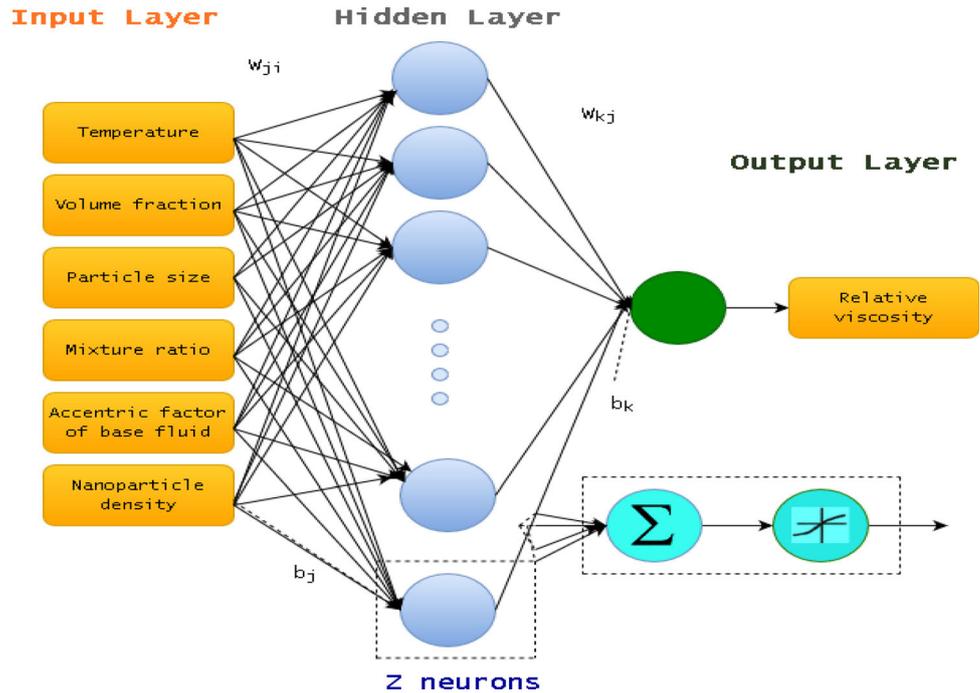
where the variance of the noise is represented as the  $\sigma_{noise}^2$

Gaussian process regression (GPR) utilize the Bayesian method which assumes a priori function values that behaves based on [81]:

$$p(f|x_1, x_2, \dots, x_n) = \mathcal{N}(0, K) \tag{5}$$

where the  $[f_1, f_2, \dots, f_n]^T$  is a vector of latent function values  $f_i = f(x_i)$  and  $K$  is a covariance matrix. The GPR function resolves the latent function values  $f_i$  as random values, indexed by the corresponding input. Considering

**Fig. 3** Hybrid nanofluid relative viscosity ANN model architecture



the D dataset, the input parameters are  $x_i$ , and  $y_i$  as the target vector, where the values are sample points. The process of regression involves creating a new input  $X^*$ , to achieve the predicted distribution for the corresponding values of the observed  $y^*$  based on the D dataset. The mean function  $m_n(x)$  defines the Gaussian process  $f(x)$ .

$$m_n(x) = E(f(x)) \tag{6}$$

The covariance  $k(x, x^i)$  function is explained using Eq. 7

$$k(x, x^i) = E(f(x) - m(x))(f(x^i) - m(x^i)) \tag{7}$$

Equation 8 expresses the Gaussian process

$$f(x) \sim \text{GPR}(m(x), k(x, x^i)) \tag{8}$$

In several GPR applications, it is a common practice to model the targets ‘y’ to be noisy realizations of the GPR, where the noise is usually parameterized to be zero, with positive noise covariance values expressed as  $\sigma_{noise}^2$

$$y_i = f(x_i) + \varepsilon \tag{9}$$

where the Gaussian distribution noise is expressed as  $\varepsilon$ . The  $\sigma_{noise}^2$  is the hyperparameter that can be optimized.

A limitation of the GPR mode is the restrictive modeling assumptions for complex data sets. The computational complexity of Gaussian process regression scales poorly with the amount of training data, which is a well-known issue [81]. However, in this study, optimization of the hyperparameters will be utilized in resolving this issue. Contreras-Reyes et al. [82] implemented a nonlinear regression analysis for mixed model by considering

additive random errors, but assumed a priori Student-t distribution.

### 2.3.1 Optimizable Gaussian process regression (O-GPR)

Despite that the GPR models are non-parametric, their hyperparameters such as length scales which significantly affect the accuracy of their predictions can be optimized [83]. This helps to maximize the predictive out-of-sample performance. This optimization, just like other supervised learning models, can be done using the gradient methods [84]. The reason for this optimization is for marginal likelihood [84]. In GPR models, however, the marginal likelihood has exponential properties, therefore, maximization log-likelihood is used for the optimization, for retrieving an analytic gradient update. Since the marginal log-likelihood function is a strictly monotonic transformation of the marginal likelihood function, the set of hyperparameters that maximizes the marginal log-likelihood will also maximize the marginal likelihood. The parameterized marginal log-likelihood is expressed by (parameterized by a set of hyperparameters  $\theta$ )

$$\begin{aligned} \log p(y|X; \theta) = & -\frac{1}{2}y^T [K(X, X) + \sigma_n^2]^{-1}y \\ & -\frac{1}{2} \log [K(X, X) + \sigma_n^2] - \log(2 * \pi) \end{aligned} \tag{10}$$

In optimizing the hyperparameters of the GPR model, the derivatives of the marginal log-likelihood are computed with respect to  $\theta$ .

$$\begin{aligned} \frac{\delta}{\delta\theta} \log p(y|X; \theta) &= \frac{1}{2} y^T K(X, X)^{-1} \frac{\delta K(X, X)}{\delta\theta} K(K, K)^{-1} y \\ &\quad - \frac{1}{2} \text{tr} \left( K(X, X)^{-1} \frac{\delta K(X, X)}{\delta\theta} \right) \\ \frac{\delta}{\delta\theta} \log p(y|X; \theta) &= \frac{1}{2} \text{tr} \left( \left( \alpha \alpha^T - K(X, X)^{-1} \right) \frac{\delta K(X, X)}{\delta\theta} \right), \\ \alpha &= K(X, X)^{-1} y \end{aligned} \tag{11}$$

The hyperparameters of the GPR are then updated using the derivatives of the gradient ascent methods. The gradient updates are in the form [85]:

$$\theta^{(i+1)} = \theta^{(i)} + \eta \nabla_{\theta^{(i)}} \log p(y|X; \theta^{(i)}) \tag{12}$$

$$\theta^{(i+1)} = \theta^{(i)} + \frac{\eta}{2} \text{tr} \left( \left( \alpha \alpha^T - K(X, X)^{-1} \right) \frac{\delta K(X, X)}{\delta\theta^{(i)}} \right) \tag{13}$$

After optimization of the GPR hyperparameters on the training data set (X, y), the GPR model will perform prediction on the test data set X\*. Furthermore, the kernel function or the covariance function explains the similarity of the data. Section 3.3 explains the different kernels with which the optimization process of the GP was carried out.

The partial dependence (PDP) and individual conditional expectation (ICE) are used for estimating the predictors in the GPR modeling.

### 2.3.2 Partial dependence plot

In a trained regression model, partial dependency represents the links between predictor variables and expected responses. Partial dependency calculates the partial dependence of projected responses on a selection of predictor variables.

Consider partial dependence on a subset of the entire predictor variable set where X = x<sub>1</sub>, x<sub>2</sub>, ... x<sub>m</sub>, and X<sup>S</sup> = x<sup>S1</sup> or X<sup>S</sup> = x<sup>S1</sup>, x<sup>S2</sup>. A subset X<sup>S</sup> contains one or two variables: X<sup>S</sup> = x<sup>S1</sup> or X<sup>S</sup> = x<sup>S1</sup>, x<sup>S2</sup>. Let X<sup>C</sup> represent the complement of X<sup>S</sup> in X. All variables in X affect the predicted response f(X):

$$f(X) = f(X^S, X^C) \tag{14}$$

The predicted values for X<sup>C</sup> defines the partial dependency of expected responses on X<sup>S</sup>:

$$f^S(X^S) = E_C[f(X^S, X^C)] = \int f(X^S, X^C) p_C(X^C) dX^C \tag{15}$$

where p<sub>C</sub>(X<sup>C</sup>) is the marginal probability of X<sup>C</sup>, that is p<sub>C</sub>(X<sup>C</sup>) ≈ P(X<sup>S</sup>, X<sup>C</sup>) dX<sup>S</sup>. Assuming that each observation is equally likely and that there is reliance between X<sup>S</sup> and

X<sup>C</sup>, as well as the interactions between X<sup>S</sup> and X<sup>C</sup> is not strong, plot Partial Reliance estimates the partial dependence as follows:

$$f^S(X^S) \approx \frac{1}{N} \sum_{i=1}^N f^{X^S, X_i^C} \tag{16}$$

where N is the number of observations and X<sub>i</sub> = (X<sub>i</sub><sup>S</sup>, X<sub>i</sub><sup>C</sup>) is the i<sup>th</sup> observation.

### 2.3.3 Individual conditional expectation (ICE)

As an extension of partial dependence, an individual conditional expectation (ICE) represents the link between a predictor variable and the projected responses for each observation. While partial dependency depicts the overall link between predictor variables and expected responses, an ICE plot disaggregates the averaged data and depicts individual dependency for each of the predictor factors.

For each observation, Partial Dependence generates an ICE plot. A set of ICE plots can be used to study partial dependence heterogeneities arising from distinct observations. With the input option Data, Partial Dependence can also construct ICE charts with any predictor data.

Consider an ICE plot for a given predictor variable x<sup>S</sup> with a given observation X<sub>i</sub><sup>C</sup>, where X<sup>S</sup> = x<sup>S</sup>, X<sup>C</sup> is the complementary set of X<sup>S</sup> in the whole variable set X, and X<sub>i</sub> = (X<sub>i</sub><sup>S</sup>, X<sub>i</sub><sup>C</sup>) is the i<sup>th</sup> observation, where X<sup>S</sup> = x<sup>S</sup>, X<sup>C</sup> is the complementary set of X<sup>S</sup> in the whole variable set X, and X<sub>i</sub> = X<sub>i</sub><sup>S</sup>. The summation in Eq. 17 is represented by the ICE plot:

$$f_i^S(X^S) = f(X^S, X_i^C) \tag{17}$$

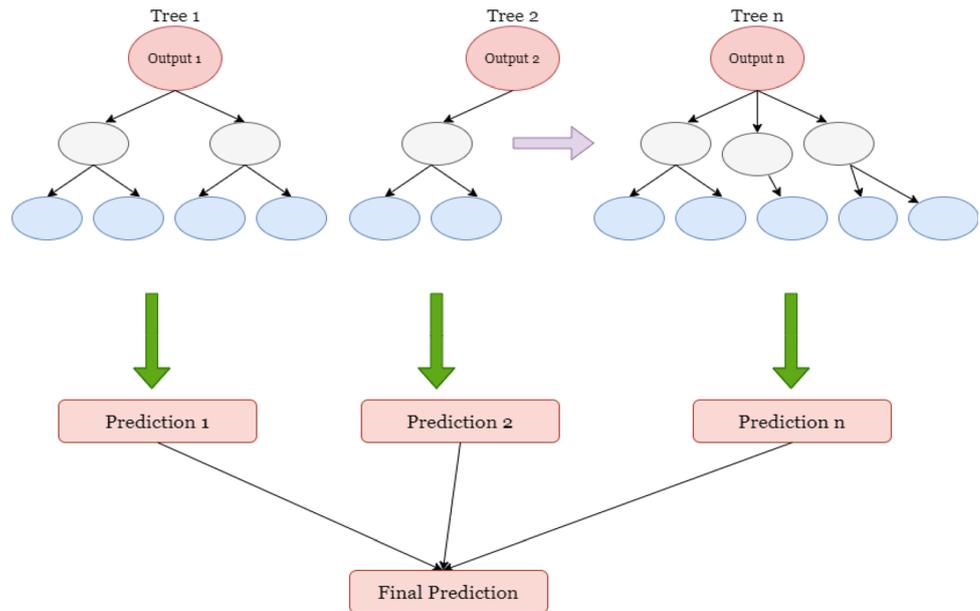
when you set 'Conditional' to 'absolute,' PlotPartialDependence plots f<sub>i</sub><sup>S</sup>(X<sup>S</sup>) for each observation i. When you set 'Conditional' to centered, plotPartialDependence creates all plots after reducing level effects caused by distinct observations.

$$f_{i,centered}^S(X^S) = f(X^S, X_i^C) - f_{min}(X^S, X_i^C) \tag{18}$$

### 2.4 Boosted tree regression (BTR)

Boosted tree regression is a combination of regression and boosted algorithms. The structure of the BTR is shown in Fig. 4. The advantage of the boosted regression is that they can model nonlinear relationships, and do not involve the removal of outliers to achieve accurate predictions. Also, BTR does not require data transformations [86]. The process of BTR development involves the introduction of a stochastic gradient procedure. There is the iterative process of fitting different tree-based models using recursive binary splits. This is done to minimize model deviance from

**Fig. 4** Boosted tree regression structure



existing trees [87]. The parameters that are defined in BTR model are the learning rate, tree complexity, number of leafs, and bag fractions. The combination of the trees to the final mode is represented by the learning rate. The tree complexity is used to decide if the interactions of the variables should be considered. The learning rate and tree complexity combine to determine the number of trees. Elith and Leathwick [86] suggested that a maximum of 1000 trees should be used when fitting models.

### 2.5 Model development

The MATLAB 2020 was used as the software environment in developing the artificial intelligence models used in this study. Upon retrieving and preparation of the dataset, preprocessing of the data was done. The preprocessing process is carried out to normalizing the dataset. The data were normalized between 0 and 1. This is done to ensure that all the variables are within the set range. Equations (22) and (23) show the mathematical computation for normalizing and denormalizing the data. The data were then split into inputs and outputs. The input variables include nanoparticle density, temperature, volume fraction, nanoparticle size, nanoparticle mixture ratio, and acentric factor. The output variable is the hybrid nanofluid relative viscosity. 700 data points were used in this study and were retrieved from 12 experimental studies. From this 700 data, the data were split randomly into 70% training and validation data and 30% testing data. The training dataset was used in estimating the optimum training parameters, and the testing dataset was used in the assessment of the developed model architecture.

$$X_n = \frac{((X_{\text{actual}} - \text{Current\_min})(\text{New}_{\text{max}} - \text{New}_{\text{min}}))}{(\text{Current}_{\text{max}} - \text{Current}_{\text{min}}) + \text{New}_{\text{min}}} \tag{19}$$

$$X_{\text{actual}} = \frac{((X_n - \text{New}_{\text{min}})(\text{Current}_{\text{max}} - \text{Current}_{\text{min}}))}{(\text{New}_{\text{max}} - \text{New}_{\text{min}}) + \text{Current}_{\text{min}}} \tag{20}$$

In Eq. (19), the  $X_n$  represents the normalized data, the  $\text{Current\_min}$  and  $\text{Current}_{\text{max}}$  is the minimum and maximum value in the dataset, while  $\text{New}_{\text{max}}$  and  $\text{New}_{\text{min}}$  is the new maximum and minimum values, respectively (1 and 0).

In assessing the predictive models developed, statistical models are used in comparing the results. The statistical measurements used in this study are the correlation coefficient ( $R^2$ ) and the mean square error (MSE) values, as shown in Eqs. (21) and (22), respectively.

$$R^2 = - \frac{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2}{\frac{1}{N} \sum_{i=1}^N (y_i - \bar{y}_i)^2} \tag{21}$$

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \tag{22}$$

where  $y_i, \hat{y}_i, \bar{y}_i$ , and  $n$  are the experimental data, predicted data, the average value of experimental data, and the number of data points, respectively.

### 2.6 Dataset

Recent studies on hybrid nanofluids were used to train and test the neural network. To improve the accuracy of the

neural network results, several nanofluids with different nanoparticles and base fluids were selected. These base fluids include 5W50 oil, 10W50 oil, SAE40 oil, AE40 oil, EG, water, and EG-water. These data sets were used so that the proposed model could account for various base fluids with different acentric factor and viscosity properties [88]. This difference increases the accuracy of the neural network across a significant range of base fluid materials. Also, particular interest was placed on hybrid nanofluids containing ZnO and MWCNT nanoparticles due to their improved stability properties [60].

The important factors that affect the relative viscosity of hybrid nanofluids based on experimental studies, as retrieved from the surveyed literature were used as the input parameters in this study. The factors are temperature (T), volume fraction (VF), density (NPD), nanoparticle size (NPS), mixture ratio (MR), and acentric factor (AF).

The data used in this study were gathered from the literature and selected based on a few variables. This is significant since the resilience and effectiveness of machine learning models are dependent on, but not limited to, data amount, data diversity, and data quality. The first criterion was that the correlation models computed in the literature (where the data is retrieved from) should have an error range of less than 1.5%. This is to ensure that the computed relative viscosity was close to the experimental values. Also, the studies which used classical models like the Einstein [70], or Wang model [89] for estimating the viscosity values were excluded when sourcing for data. This is because the classical models significantly deviate from experimental values [52]. Also, the experimental which included the different factors like the nanoparticle properties like density and size were chosen. This was done to maintain a uniformity of the various studies.

Table 2 shows the sources of the hybrid nanofluid used in the predictive models developed in this study.

$$\mu_{nf} = \mu_{nf}(T, VF, NPD_i, NPS_i, MR_i, AF) \quad i = 1, 2 \quad (23)$$

Furthermore, in this study, considering that there are two nanoparticles in the hybrid, the data preparation involved splitting the data into particle 1 and particle 2. The NP1D represents the density of nanoparticle 1, NP2D represents the density of nanoparticle 1, NP1S represents the size of nanoparticle 1, NP2S represents the size of nanoparticle 2, MR1 represents the mixture ratio of nanoparticle 1, MR2 represents the mixture ratio of nanoparticle 2. The relative viscosity is represented as CORR.

Table 3 shows that the highest number of data points was retrieved from the study carried out by Hemmat Esfe et al. [96]. It should be noted that the data used were independent of the shear rate variations, as the values that were used followed the Newtonian behavior. Table 2 also shows that the experimental studies were conducted

between volume fractions of 0 and 1%, with an exception of a study by Nabil et al. [93] which investigated the viscosity at a volume fraction of 3%. Also, considering that the stability of nanofluids is important, especially for application in solar technologies, Table 2 shows the different techniques used for ensuring stable suspensions. The descriptive analysis of the dataset is shown in Table 3

### 3 Results and model assessment criteria

In assessing the prediction accuracy of the O-GPR, BTR, ANN model, and SVR model, the performance evaluation tool used is the correlation coefficient ( $R^2$ ) and mean square error (MSE) values. For the models, the training process of the dataset was done ten (10) times and the average value is computed. In the development of the O-GPR, BTR, ANN and SVR models for prediction of the relative viscosity of the hybrid nanofluids, the input variables used are temperature (T), volume fraction (VF), density (NPD), nanoparticle size (NPS), mixture ratio (MR) and acentric factor (AF).

#### 3.1 Artificial neural network

The training of the ANN model was done using datasets which were randomly retrieved from the preprocessed dataset. The optimum ANN architecture was developed from a rigorous trial and error process using different transfer functions, the number of hidden layers, and neurons [71]. Before the training process, the dataset was normalized for computational efficiency [69]. Table 4 shows the different parameter ranges used in the ANN model development. The learning algorithms, maximum iteration, minimum gradient as well as threshold function adopted for the ANN models are highlighted in Table 4

Table 5 shows the statistical result of the training, testing, and validation data points using different ANN architectures. The training function used in the model development is the Bayesian regularization (BR) training function. The BR training function is used due to its ability to minimize overfitting problems in the training process as it takes into account the goodness of fit [68]. The BR function model has a comparatively better performance compared to other training functions [72, 73].

From Table 5, it is seen that the optimum ANN architecture was obtained from the configuration having 2 hidden layers with 10 and 30 neurons in the first and second layers, respectively. The validation dataset result for this optimum configuration showed an  $R^2$  value of 0.9975 and an MSE value of 0.0119.

A comparative analysis is further done using this optimum model configuration, on different training functions.

**Table 2** Hybrid nanofluid data used in the prediction of relative nanofluid

No	Nanoparticle	Base Fluid	Nanofluid concentration	Temperature (°C)	Stability	No of data points	References
1	Al <sub>2</sub> O <sub>3</sub> /Fe	Water	0.05%, 0.1%, and 0.2%	25–65	PH modulation	35	[90]
2	Al <sub>2</sub> O <sub>3</sub> /MWCNT	Water/EG (80:20)	0.0625, and 1%	25–50	Mixing and sonication	39	[50]
3	Al <sub>2</sub> O <sub>3</sub> /MWCNT	5W50 Oil	0.05–1%	25–50	Mixing and sonication	41	[91]
4	MWCNT/MgO	EG	0–1.0%	30–60	Ultrasonication/pH modulation	45	[48]
5	Al <sub>2</sub> O <sub>3</sub> /MWCNT	SAE40 Oil	0–1.0%	25–50	Mixing by magnetic stirrer for 2 h	51	[92]
6	TiO <sub>2</sub> /SiO <sub>2</sub>	Water/EG (60:40)	0.5–3.0%	30–80	Sonication for 90 min	51	[93]
7	ZnO/Ag	Water	0.125–2%	25–50	Surfactant (not stated)	57	[74]
8	SiO <sub>2</sub> /MWCNT	SAE40 Oil	0–2.0%	25–50	Mixing and sonication	57	[94]
9	ZnO/MWCNT	10W50 Oil	0.05–1%	5–55	Ultrasonic waves for 6 h, using 1200W ultrasonic processor	57	[95]
10	SiO <sub>2</sub> /MWCNT	AE Oil	0.0625–1.0%	25–60	Mixing and sonication	63	[62]
11	SiO <sub>2</sub> /MWCNT	SAE40 Oil	0–1.0%	25–60	Magnetic stirring for 2.5 h, the suspensions were exposed to an ultrasonic processor	63	[45]
12	ZnO/MWCNT	SAE40 Oil	0–1.0%	25–60	Magnetic stirring for 1 h, exposed to an ultrasonic processor of KND-1200-UH1 with power of 1200 W for 8 h	141	[96]

**Table 3** Descriptive statistics of predictive and target variables

	NP1D	NP2D	NP1S	NP2S	MR1	MR2	T	VF	AF	CORR
Count	700	700	700	700	700	700	700	700	700	700
Mean	0.420327	0.385824	0.615357	0.466514	0.651810	0.638929	0.494375	0.169333	0.836098	0.466759
Standard deviation	0.186022	0.180993	0.209199	0.214781	0.125097	0.117278	0.156332	0.167659	0.220202	0.107086
Minimum	0.228571	0.044595	0.250000	0.160000	0.266667	0.312500	0.062500	0.000000	0.480392	0.304905
25%	0.228571	0.374599	0.500000	0.400000	0.666667	0.625000	0.375000	0.041667	0.682073	0.398058
50%	0.370476	0.374599	0.500000	0.400000	0.666667	0.625000	0.500000	0.133333	1.000000	0.439498
75%	0.533905	0.374599	0.625000	0.520000	0.666667	0.625000	0.625000	0.250000	1.000000	0.484465
Maximum	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000

The training functions used are the Levenberg–Marquardt (LM), scaled conjugate gradient (SCG), and the gradient descent with momentum (GDM) training functions. Table 6 shows the performances of the different models.

As can be observed from Table 6, the best prediction performance is from the BR model. The LM model gives the second-best prediction accuracy with an  $R^2$  value of

0.9883, while the worst performance is seen with the GDM training function. A graphical representation of the optimum ANN model architecture is shown in Fig. 1. Also, the low MSE value of the optimum model further shows the excellence of the model for making predictions.

As can be seen in Fig. 5, all the data points are aligned along the diagonal line which shows an efficient prediction

**Table 4** Parameters for the ANN training algorithm

Parameters	Values
Maximum iteration	1000
Minimum gradient	1e-07
Training dataset	70%
Testing dataset	30%
Learning rate	0.1
Number of hidden layers	1 layer, 2 layers
Node in hidden layers	1 Hidden layer-10, 15, 20, 25/30 2 Hidden layer -5/10, 5/15, 5/25, 10/10, 10/15, 10/20, 10/25, 10/30
Learning algorithm	LM, SCG, BR, GDM
Threshold function	Logistic sigmoid – neuron activation function Purelin – activation function for the output neuron
Performance metric	R <sup>2</sup> , MSE, RMSE, MAE

**Table 5** Statistical result of ANN configuration

Hidden Layers	Neurons	Training		Testing		Validation	
1	5	0.00122963	0.980915	0.0526	0.9896	0.0327	0.9803
	10	0.000252779	0.996192	0.0294	0.9968	0.0170	0.9950
	15	0.000130475	0.997945	0.0241	0.9978	0.0149	0.9961
	20	0.00009800	0.998477	0.0182	0.9988	0.0157	0.9957
	25	0.000111477	0.998249	0.0234	0.9979	0.0123	0.9973
	30	0.000126951	0.9980123	0.0213	0.9983	0.0144	0.9963
2	5–10	0.00035925	0.9945	0.0174	0.9989	0.0258	0.9881
	5–15	0.00018373	0.9972	0.0271	0.9973	0.0158	0.9956
	5–25	0.00015144	0.9976	0.0254	0.9976	0.0147	0.9962
	10–10	0.00016053	0.9975	0.0242	0.9978	0.0150	0.9960
	10–15	0.00017828	0.9973	0.0198	0.9985	0.0176	0.9944
	10–20	0.0001087	0.9984	0.0163	0.9990	0.0144	0.9964
	10–25	0.0001819	0.9982	0.0167	0.9990	0.0150	0.9960
	10–30	0.00011027	0.9983	0.0235	0.9979	0.0119	0.9975

**Table 6** Statistical result of ANN configuration with different transfer functions (using optimum hidden layers and neurons)

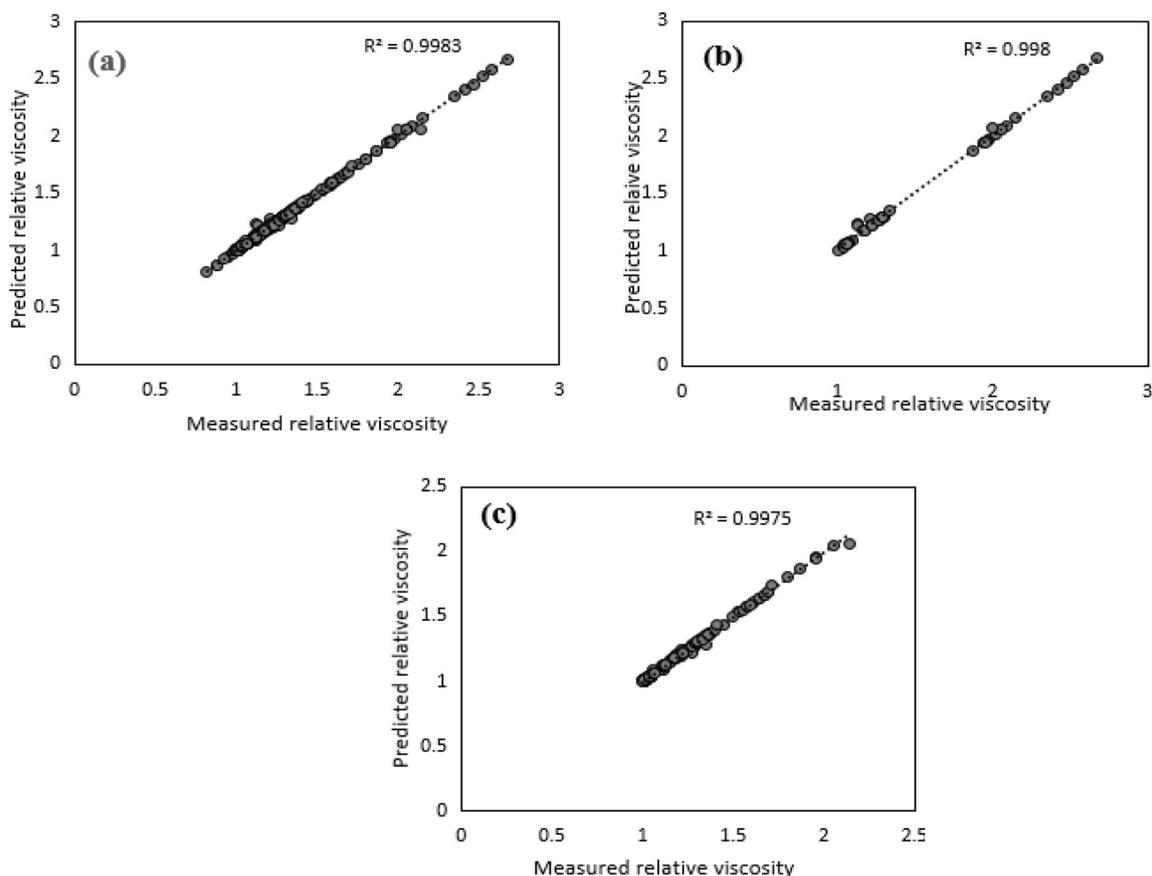
Transfer Function	Neurons	Training		Testing		Validation	
		MSE	R <sup>2</sup>	MSE	R <sup>2</sup>	MSE	R <sup>2</sup>
BR	10–30	0.00011027	0.9983	0.00055225	0.9979	0.00014161	0.9975
LM	10–30	0.0005998	0.9908	0.00101761	0.9963	0.00065025	0.9883
SCG	10–30	0.0119	0.8147	0.02313441	0.8718	0.00620944	0.8680
GDM	10–30	0.3619	0.0302	0.197136	0.0046	0.0015460624	0.3351

analysis. The R<sup>2</sup> value for the training, testing, and validation data are 0.9983, 0.9980, and 0.9975, respectively. The high correlation coefficient of the training, testing, and validation data shows that the model can be used to accurately make predictions of the relative viscosity of hybrid nanofluids.

Based on the optimum ANN model developed, a universal formula is derived for obtaining the viscosity of hybrid nanofluids Eq. (24).

$$\sum_{i=1}^{30} w_{i1}^3 (1/1 + \exp(-(\sum_{j=1}^{10} w_{ji}^2 (1/1 + \exp(-(\sum_{k=1}^{11} w_{kj}^1 u_k + \beta_j^1))) + \beta_i^2))) + \beta_1^3 \quad (24)$$

where the bias weights are given as  $\beta_j^k$  and  $w_{ji}^k$  is the link weights for the neuron  $j$  in layer  $k$ .  $u_k$  is the input variable matrix for the six input parameters. A supplementary file containing the weights and bias of the optimum ANN model architecture is attached.



**Fig. 5** Measured relative viscosity vs ANN predicted relative viscosity for (a) Training data points (b) Testing data points (c) Validation data points

### 3.2 Support vector machine

The MATLAB regression learner toolbox was used for developing the model. As stated in the previous section, the dataset was split into 70% training and 30% validation dataset. The generalization capabilities of the SVR model, which involves the effective prediction of the data validation data in the training phase, is affected by the type of kernel function, gamma of the kernel function, the bias of the kernel function, and the degree of the polynomial kernel function. In this study, these factors are tuned several times to get the optimum SVR model for accurate prediction. The polynomial function was selected for this study as it gave the best performance. The cubic polynomial was comparatively better in performance than the linear, RBF, and sigmoid functions. Figure 2 shows the comparative performance of the different kernel functions. The cross-validation used is the K-fold CV, and this infers the generalization ability [53]. In this study, the CV was set to 5, and the epsilon value which describes the error between the predicted and measured value in a high-dimensional space [54] was optimized to a value of 0.0741. Table 7 shows the optimum parameters for SVR modeling.

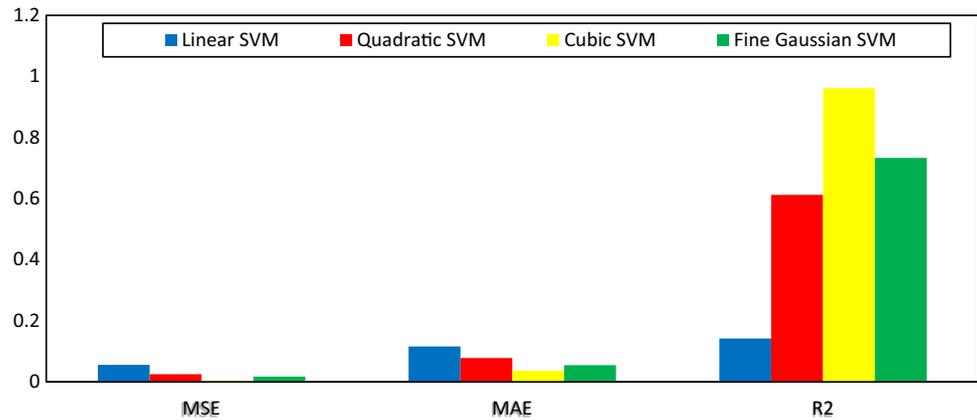
**Table 7** Optimum parameters for SVR modeling

Model parameters	Optimal values
Function	Cubic SVM
Kernel scale	0.9112
Bias	1.4719
K folds	5
Epsilon ( $\epsilon$ )	0.0741

Figure 6 shows that the cubic kernel gave the best prediction. Table 8 shows the performance of the training, testing, and validation data retrieved from the SVR model. The high  $R^2$  value of 0.9636 gotten from the validation data showed an efficient prediction accuracy. Also, the low MSE values for the training, testing, and validation data confirm the high prediction performance of the optimum SVR model developed in this study.

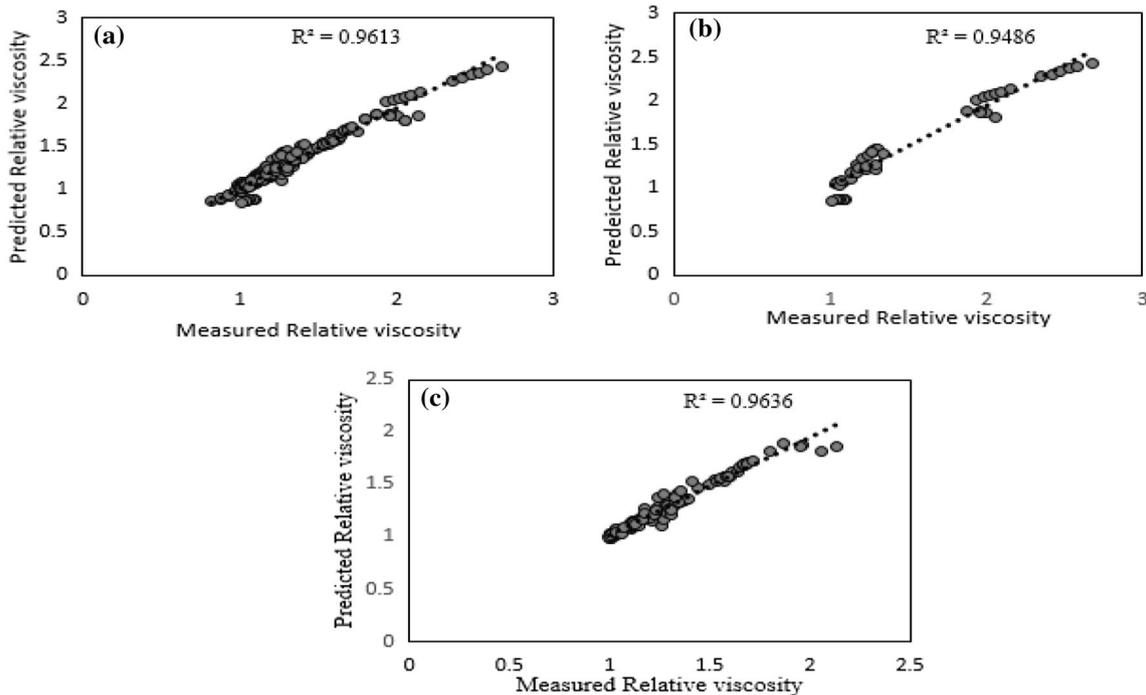
Figure 7 shows the cross-plot between the measure relative viscosity and the predicted relative viscosity by the SVR model. As seen in Fig. 7, the clustering of the data points along the straight line shows that there is a high agreement between the measured and the predicted values.

**Fig. 6** Comparative statistical result of different kernel functions in SVR (training data points)



**Table 8** Optimum statistical results for SVR modeling

Parameters	Training dataset	Testing dataset (MSE)	Validation dataset (MSE)
MSE	0.0037552	0.01404225	0.00210681
R <sup>2</sup>	0.9613	0.9486	0.9636



**Fig. 7** Measured relative viscosity vs SVR predicted relative viscosity for (a) Training data points (b) Testing data points (c) Validation data points

The  $R^2$  value of the training, testing, and validation data showed that the model has a high accuracy for the prediction of the relative viscosity of the hybrid nanofluids.

### 3.3 Optimized Gaussian process regression

In Gaussian process regression, the response is modeled using a probability distribution over a space of functions.

The flexibility of the presets in the Model Type gallery is automatically chosen to give a small training error and, simultaneously, protection against overfitting. The MATLAB 2020 is the simulation environment used in developing the O-GPR model. The nine (10) features used in developing the model are labeled NP1D, NP2D, NP1S, NP2S, MR1, MR2, TEMP, VF, AF, and CORR, as explained in Eq. 26. In this study, an algorithm is designed

that searches among the following kernel functions, namely: Nonisotropic Rational Quadratic, Isotropic Rational Quadratic, Nonisotropic Squared Exponential, Isotropic Squared Exponential, Nonisotropic Matern 5/2, Isotropic Matern 5/2, Nonisotropic Matern 3/2, Isotropic Matern 3/2, Nonisotropic Exponential, and Isotropic Exponential.

Using the data, the Gaussian process regression model is trained. The response variable includes interactions between predictor variables. To establish the relationship between a feature and the predicted responses for each observation, the ICE plot is implemented. The cumulative effect of the selected feature can be examined by using the offset plots to start from zero, using the MATLAB 'Conditional' setting. Table 9 shows the optimizer functions used in developing the O-GPR model.

The predictor variables are also compared to ascertain their importance in the prediction process. Also, the estimates of predictor importance are analyzed. The predictor importance function summarizes the importance of a predictor with a single value. This function sums changes in the mean squared error (MSE) due to splits on every predictor and then divides the sum by the number of branch nodes. The predictors are ascribed 1, 2, 3, 4, 5, 6, 7, 8, 9 to NP1D, NP2D, NP1S, NP2S, MR1, MR2, TEMP, VF, AF, respectively.

Table 10 also shows the predictor ranks in terms of their importance in estimating the target variable (Relative viscosity). Table 10 shows that the 8th variable (volume fraction) has the most impact on Relative viscosity according to predictor importance. This corroborates with experimental results, that explain the significant effect of volume fraction on relative viscosity [12, 55, 56]. The PDP of the 8th Variable also shows that CORR has high partial dependence on the 8th Variable. The 2nd variable (nanoparticle density) also shows high importance in the prediction of relative viscosity. This also supports results in the literature, as a study by Corcione et al. [12] developed a correlation formula for viscosity prediction using the nanoparticle density and volume fraction, which gave excellent prediction based on the small deviation recorded. The 6th variable (Mixture ratio) has the least impact on relative viscosity according to predictor importance.

**Table 9** Optimizer functions

Optimizer options	
Optimizer	Bayesian optimization
Iterations	30
Training time limit	False

The PDP of 6th also shows that relative viscosity does not change depending on the feature. The PDP of 5th also shows that relative viscosity does not change much depending on the feature.

The algorithm kernel scale searches among real values in the range [0.075–75], sigma between [0.0001–2.5347], and standardization between true and false as estimated. This mode is important as standardizing removes the dependence on arbitrary scales in the predictors and generally improves performance. Standardizing the predictors transforms them so that they have a mean of 0 and standard deviation of 1. A value of 0.21431 is assigned as an initial Kernel scale parameter developed via a heuristic procedure.

In this study, hyperparameter tuning is performed by using Bayesian optimization. The goal of Bayesian optimization, and optimization in general, is to find a point that minimizes an objective function (loss function/MSE). Figure 8 shows the error index. Figure 8 shows light blue points representing the estimated minimum MSE, and dark blue point corresponds to the observed minimum MSE which is drawn out during the optimization process. For holdout validation, the score is the RMSE on the held-out observations. An RMSE of 0.022056 is obtained. The Coefficient of determination (R-squared) value of 0.999998 is obtained, which shows a significantly excellent accuracy, as this value is expected to be closer to 1. The mean square error MSE value is 0.00048645 which is quite small and desired. The mean absolute error MAE just like RMSE, is desired to be a small value; however, they are less sensitive to outliers. An MAE of 0.013111 is recorded.

Figures 9 and 10 show the predicted relative viscosity values for training (validation) and testing sets show close values to the measured values. The R-squared values recorded for the training and testing sets are 0.999989 and 0.999998, respectively, as shown in Table 11.

### 3.4 Boosted tree regression (BTR)

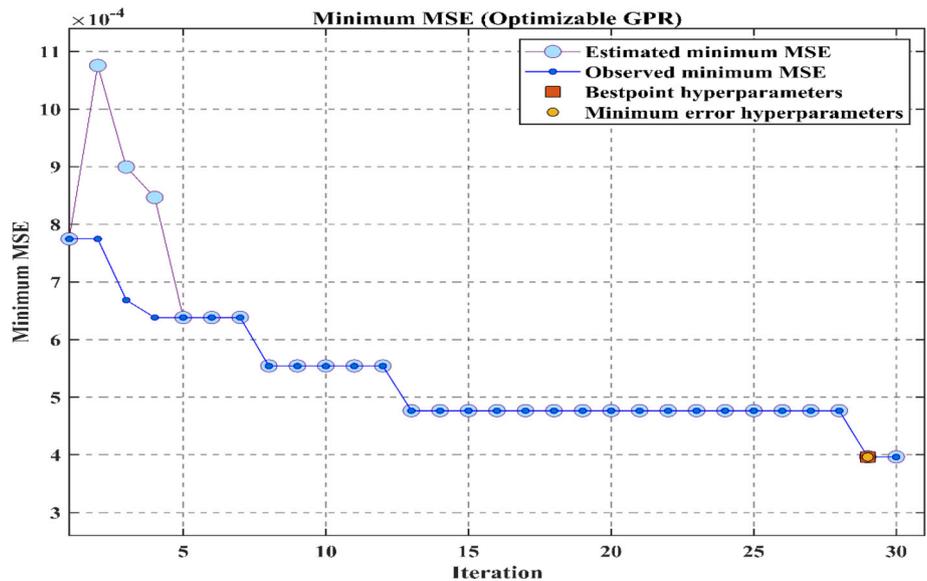
The BTR model is developed in the MATLAB 2020 environment. Table 12 shows the parameters defined in developing the BTR model.

Figures 11 and 12 which depict the predicted relative viscosity values for training (validation) and testing sets, respectively, show close values to the measured values. The R-squared values recorded for the training and testing sets are 0.887874 and 0.867569, respectively, as shown in Table 13. The MSE result recorded for the training and test model is 0.007943 and 0.0078502, respectively. The BTR is seen to be comparatively less efficient than the other machine learning tools used in this study, and this can be attributed to its over-sensitivity to outliers as every classifier is required to rectify the flaws in the predecessors.

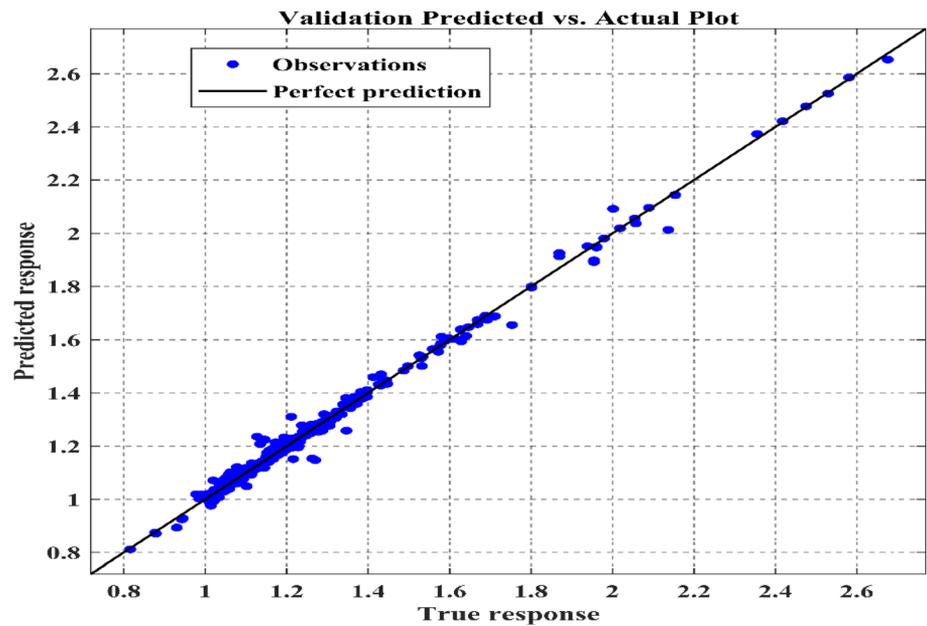
**Table 10** Predictor ranking

Predictor	1	2	3	4	5	6	7	8	9
Importance	8.58e-06	2.24e-05	2.16e-06	9.90e-07	1.28e-07	4.34e-09	1.34e-06	2.46e-05	3.71e-06
Rank	3rd	2nd	5th	7th	8th	9th	6th	1st	4th

**Fig. 8** Error index against iterations of the O-GPR model



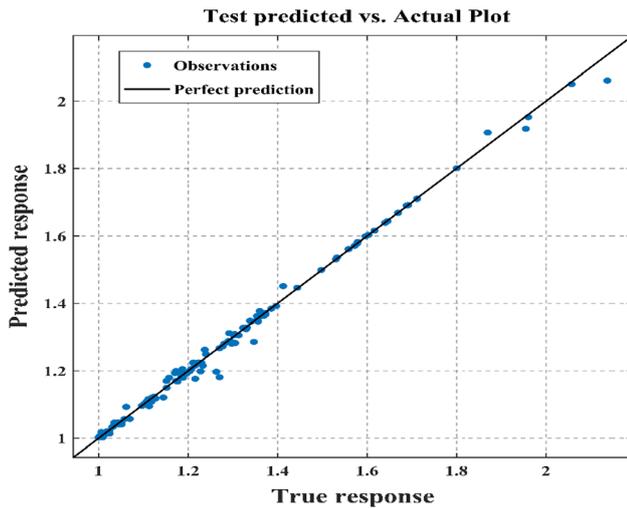
**Fig. 9** Predicted versus experimental values of training (validation) relative viscosity dataset of the O-GPR model



Furthermore, the BTR can overemphasize outliers and cause overfitting. It is also of worthy note that tree-based models aren't meant to function with very few features [57, 58].

### 3.5 Comparison of models used in this study

Table 14 shows a comparison of the statistical result of the O-GPR, BTR, ANN, and SVR models. The models showed



**Fig. 10** Predicted versus experimental dataset of test of relative viscosity dataset of the O-GPR model

a high prediction accuracy based on their high  $R^2$  and low MSE values. The O-GPR model outperformed the other models, which goes to show that a better prediction for the viscosity of hybrid nanofluids will be achieved using the O-GPR model developed in this study.

### 3.6 Model validity

This section compares the ability of the machine learning algorithms used in this study with some theoretical models used in the literature to estimate the relative viscosity of hybrid nanofluids.

Theoretical models have been developed by several researchers for determining the viscosity of nanofluids. It is observed that most of the models developed for the viscosity of nanofluids do not include the temperature parameter or many of the parameters used in developing

**Table 12** Parameters for BTR development

Parameters	Values
Minimum leaf size	8
Number of learners	30
Learning rate	0.1

the model used in this study [59]. Yan et al. [60] stated that the nanofluid viscosity is greatly affected by the temperature. The Brinkman [61], Wang et al. [32], Cheng et al. [44], and Ludgren [62] models were compared with the optimum AI models developed.

The Brinkman model was developed as an expansion of the Einstein model [18], which was used in determining the viscosity of a mixture of solid particles suspended in a liquid. The Brinkman model as shown in Eq. (25) was used for larger volume fractions.

$$\frac{\mu_{nf}}{\mu_{bf}} = (1 - \varphi)^{2.5} \tag{25}$$

Also, a model was proposed by Wang et al. [32], to determine, the dynamic viscosity of nanofluids is shown in Eq. (26).

$$\frac{\mu_{nf}}{\mu_{bf}} = (1 + 7.3\varphi + 123\varphi^2) \tag{26}$$

Another model was proposed by Ludgren [62], using the Taylor series form

$$\frac{\mu_{nf}}{\mu_{bf}} = \frac{1}{(1 - 2.5\varphi)} = (1 + 2.5\varphi + 6.25\varphi^2) \tag{27}$$

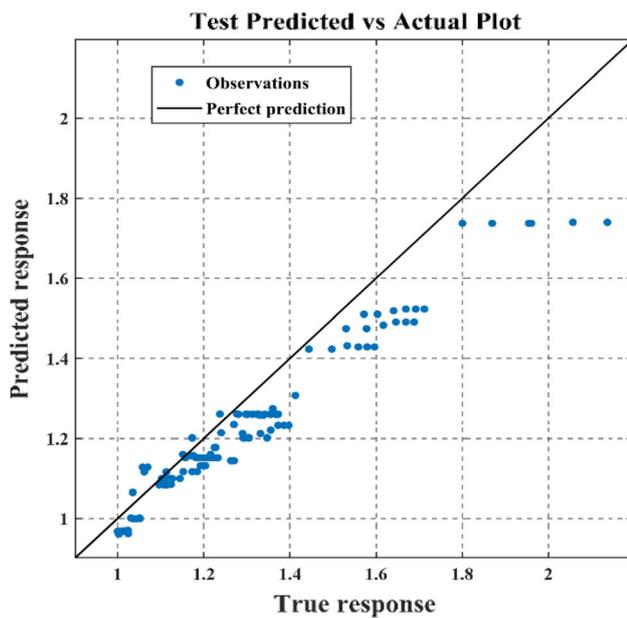
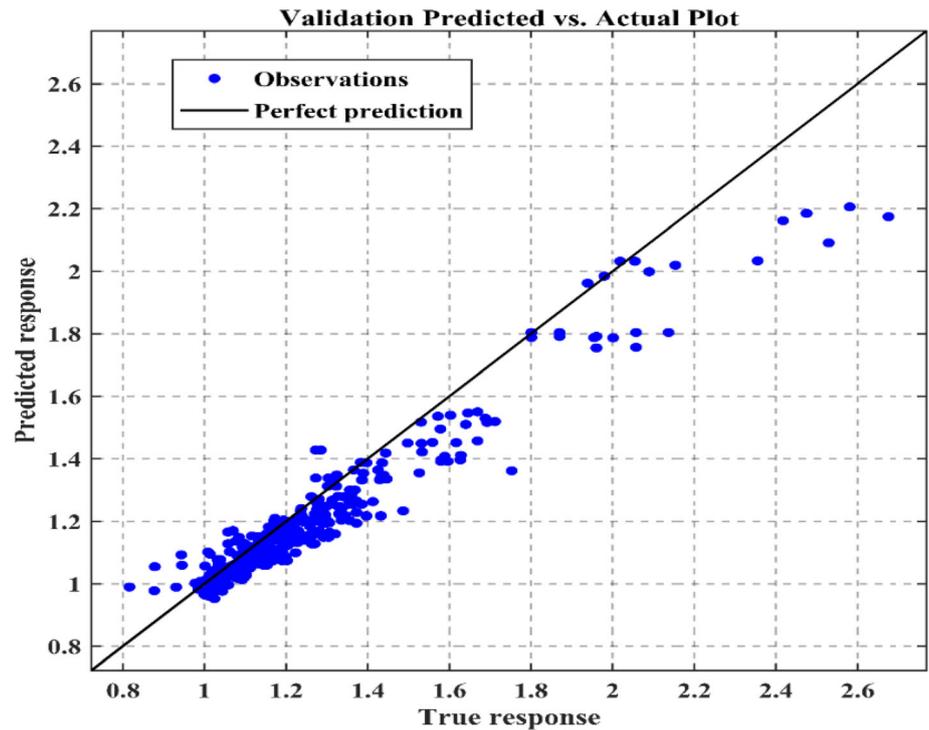
where  $\frac{\mu_{nf}}{\mu_{bf}}$  is the relative viscosity and  $\varphi$  is the volume fraction of the nanofluid.

The theoretical models were used to compute the relative viscosity of the  $Al_2O_3/MWCNT$  hybrid nanofluid [34] and compared with the predicted values of the O-GPR,

**Table 11** Optimizable GPR results

Phase	Description	
Model type	Signal standard deviation	Optimizable GPR 0.17923
Training (Validation) result	Observations	490 (70%)
	Features	9
	RMSE	0.022056
	R-Squared	0.999989
	MSE	0.00039607
	MAE	0.01107
	Training time	3152 s
Testing Result	Observations	210 (30%)
	RMSE	0.015975
	MSE	0.0002552
	MAE	0.0084404
	R-Squared	0.999998

**Fig. 11** Training (validation) predicted VS experimental values of relative viscosity of BTR model



**Fig. 12** Test predicted versus experimental values of relative viscosity of BTR model

BTR, ANN and SVR models as seen in Fig. 13. The relative viscosity of the Al<sub>2</sub>O<sub>3</sub>/MWCNT hybrid nanofluid was measured and predicted at 25 °C and volume concentration of 0.0625, 0.125, 0.25, 0.5, 0.75, and 1% [34]. This experimental nanofluid was retrieved from a study done by Afshari et al. [34]. The purpose of this analysis is to compare the robustness of the machine learning models

**Table 13** Statistical error of BTR model

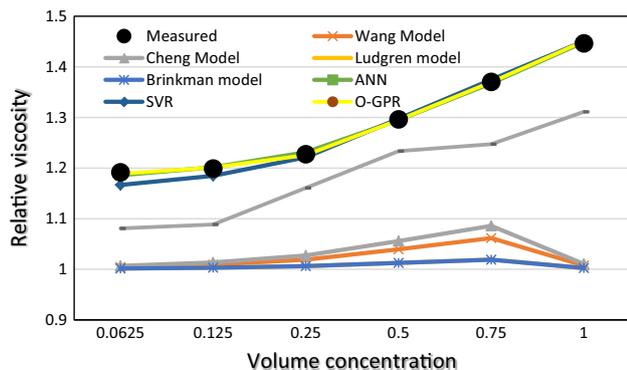
	Training (Validation)	Testing
RMSE	0.089124	0.088601
MSE	0.007943	0.0078502
MAE	0.063919	0.067881
R-Squared	0.887874	0.867569

developed in this study with other classical models for viscosity estimations, across varying volume concentrations.

Figure 13 shows a high deviation of the theoretical models in estimating the relative viscosity. The inefficiency of the theoretical models to effectively predict the relative viscosity of the hybrid nanofluid is because the models only accounted for the volume fraction as a variable in its computation. It is therefore seen that despite the significant effect of volume fraction on the relative viscosity behavior of nanofluid, other parameters are important for their accurate estimation. Figure 13 shows that the Cheng et al. [44] model predicts the trend better than the other models. This can be attributed to the higher coefficient of the second-order volume fraction. This is corroborated by a study carried out by Mahdi et al. [56]. The relative viscosity values of the O-GPR, BTR, ANN, and SVR models are closer to the measured values. The maximum deviation of the O-GPR, BTR, ANN, and SVR

**Table 14** Performance of proposed machine learning algorithms

Machine learning algorithms	O-GPR	BTR	ANN	SVR
MSE	0.0002552	0.0078502	0.00014161	0.00210681
$R^2$	0.999998	0.867569	0.997578	0.963667
Model rank	1	4	2	3

**Fig. 13** Comparison of measured values of relative viscosity with predicted values using different models for  $\text{Al}_2\text{O}_3/\text{MWCNT}$  hybrid nanofluid, with a mixture ratio of 75:25%, at  $T = 25^\circ\text{C}$ 

models are 0.286, 9.35, 0.497, and 0.482, respectively. The study of Mehdi et al. [49] corroborates the efficacy of the GPR model as their study also showed excellent prediction accuracy in their SHC prediction model. This study only considered Newtonian nanofluid, so the shear rate was not considered as a parameter.

### 3.7 Comparative analysis with other studies/ results on viscosity prediction of nanofluids

As stated in earlier sections, improvement in prediction accuracy of thermophysical properties is important in the discourse of their application to solar energy technologies as heat transfer fluids. The result retrieved in the study being compared with that retrieved by other authors.

From Table 15, it is seen that this present study utilized more diverse input variables (6) in the prediction analysis, as compared to other studies. Considering that machine learning algorithms are data-driven, the inclusion of more diverse variables ensures better mapping of the predictors to the target values. Table 15 also shows that the R-squared value recorded in this study is the closest to the desired value of 1, which shows excellent prediction accuracy of the viscosity of hybrid nanofluids. The MSE value of 0.0002552 in this study is also the lowest seen in works of literature.

## 4 Implication on solar energy system application/future remarks

The application of nanofluids for solar energy technologies has been shown to improve their efficiencies. Furthermore, another advantage of nanofluid-based solar systems is cost-effectiveness, which a study by Taylor et al. [117] showed. In their study, graphite nanofluids were used for a 10–100 MW solar power tower system. Their result showed a 10% increase in efficiency in utilizing nanofluids in the receiver section of the system, which would yield an estimated 3.5 million dollars per year as revenue. However, as earlier stated, the utilization of nanofluids for solar energy technologies is dependent on trade-offs between the different thermophysical and rheological properties of thermal conductivity, viscosity, density, and specific heat capacity. Improving thermal conductivity, which is usually achieved by increasing temperature and volume concentration, is desirable. However, an increase of volume fraction beyond a certain threshold results in a net negative in the viscosity of the fluids [118]. This gives a strong argument that efficiently ascertaining the optimum viscosity plays a significant role in the design of nanofluid-based solar energy technologies. Furthermore, the design of synthesis for the nanofluid study is usually expensive, especially for hybrid nanofluids, therefore, there is a need for robust and efficient AI models that would be able to accurately estimate this property. This in turn will make the design of nanofluid-based solar systems more cost-effective. The growth in the field of AI has afforded the leverage to utilize novel machine learning algorithms to more accurately estimate the viscosity of nanofluids. This study has shown that the application of Optimizable Gaussian process regression yields a very high prediction accuracy for viscosity estimation.

From a futuristic perspective, developing safe and sustainable energy solutions to meet increasing energy demands has become more demanding due to growing population and a need to cut fossil fuel usage. Solar collector systems have shown to be an effective alternative energy producer. Therefore, more research into increasing its implementation in global energy mix is key to solving more energy problems. The practical application of nanofluids for solar thermal systems is still in its infancy stage despite the knowledge of their improved efficiency. This is attributed to factors like cost, synthesis methods,

**Table 15** Comparison with other studies

References	Nanofluids	Data size	Input variables	Machine learning	Statistical error index	R-squared
Karimi et al. [74]	Al <sub>2</sub> O <sub>3</sub> , CuO, TiO <sub>2</sub> , SiO <sub>2</sub>	114	Temperature, volume fraction, Particle size, base fluid viscosity	GA-NN	MARE – 2.48%	0.998001
Li et al. [63]	Al <sub>2</sub> O <sub>3</sub>	48	Temperature, Volume fraction	ANN	MSE – 0.00837	0.99972
Ma et al. [64]	Al <sub>2</sub> O <sub>3</sub> -CuO SiO <sub>2</sub> -TiO <sub>2</sub>	99	Temperature, Mixture ratio	ANN	–	0.9755
Hemmati-Sarapardeh et al. [65]	Al <sub>2</sub> O <sub>3</sub> , CuO, SiO <sub>2</sub> , TiO <sub>2</sub> , SiC, Fe <sub>3</sub> O <sub>4</sub> , MgO, Mg(OH) <sub>2</sub> , CO <sub>3</sub> O <sub>4</sub> , Nanodiamonds, ZnO	3144	Temperature, volume fraction, Particle size, base fluid viscosity	MLP, LSSM	MSE = 0.008649 AARE = 3.95%	–
Gholizadeh et al. [66]	Al <sub>2</sub> O <sub>3</sub> , CuO, SiO <sub>2</sub> , TiO <sub>2</sub> , SiC, Fe <sub>3</sub> O <sub>4</sub> , MgO, Mg(OH) <sub>2</sub> , CO <sub>3</sub> O <sub>4</sub> , Nanodiamonds, ZnO, CNT, MWCNT, COOH	2890	Temperature, volume fraction, Particle size, basefluid viscosity, Density of nanoparticle	Random Forest, MLP, SVR	MSE = 0.019321	0.978121
Jamei et al. [67]	Al <sub>2</sub> O <sub>3</sub> -MWCNT SiC- TiO <sub>2</sub> , MWCNT-MgO, MWCNTs- ZnO, MWCNT- SiO <sub>2</sub>	679	Temperature, volume fraction, Particle size, basefluid viscosity, Density of nanoparticle	MGGP, MLR	0.0025	0.982081
Present Study	MWCNT-MgO, Al <sub>2</sub> O <sub>3</sub> -MWCNT, ZnO-Ag, TiO <sub>2</sub> - SiO <sub>2</sub> , Al <sub>2</sub> O <sub>3</sub> /Fe, SiO <sub>2</sub> -MWCNT	700	Temperature, volume fraction, Particle size, basefluid viscosity, Density of nanoparticle, Mixture ratio	O-GPR, ANN, SVR, BTR	0.0002552	0.999998

and a few experimental nanofluid-based solar designs. For more implementation of nanofluid-based solar system designs, the numerical estimation of the thermophysical properties of nanofluids has to be developed and utilized.

The authors also suggest that solar energy designers should keep updated with novel machine learning tools, as their utilization for giving more accurate prediction of thermophysical properties of nanofluids is instrumental in the search for more cost-effective efficient solar energy technologies.

## 5 Conclusion

This study shows the applicability of efficiently predicting the relative viscosity of hybrid nanofluid. The proposed O-GPR, BTR, ANN, and SVR models were built using six input parameters namely nanoparticle density, nanoparticle size, mixture ratio, acentric factor, temperature, and volume fraction. 700 data points were retrieved from 12 experimental studies on hybrid nanofluids. The optimum O-GPR, BTR, ANN, and SVR models were developed from varying several parameters in their architecture. Furthermore, the study also compares the prediction result of the AI models with theoretical models for predicting viscosity. Based on the statistical result obtained, the following conclusions are made.

- The O-GPR model has a comparatively better prediction performance of the relative viscosity of hybrid nanofluids. A coefficient of determination of 0.999998 was recorded.
- The O-GPR model has a better prediction accuracy of the relative viscosity when compared with other BTR, ANN, and SVR models, as the lowest MSE value of 0.002552 was recorded.
- The O-GPR, BTR, ANN, and SVR models have better performance than the theoretical models.
- The MSE result recorded for the training and test model for the BTR model is 0.007943 and 0.0078502, respectively
- The MSE result recorded for the training and test model for the SVR model is 0.0037552 and 0.01404225, respectively
- The maximum deviation of the O-GPR, BTR, ANN, and SVR models used in estimating the Al<sub>2</sub>O<sub>3</sub>/MWCNT hybrid nanofluid are 0.286, 9.35, 0.497, and 0.482, respectively
- The R<sup>2</sup> values for the O-GPR, BTR, ANN, and SVR models are 0.999998, 0.867569, 0.997578, and 0.963667
- Theoretical models are not accurate predictors of relative viscosity as they do not consider other parameters that affect the viscosity of the nanofluids.
- The design of synthesis for the nanofluid study is usually expensive, especially for hybrid nanofluids; therefore, there is a need for robust and efficient AI

models that would be able to accurately estimate this property, which will make the design of nanofluid-based solar system more cost-effective. This study shows that the implementation of the optimizable Gaussian process regression gives excellent prediction accuracy.

## Declarations

**Conflict of 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. The authors declare the following financial interests/personal relationships which may be considered as potential competing interests.

## References

- Che Sidik NA, Mahmud Jamil M, Aziz Japar WMA, Muhammad Adamu I (2017) A review on preparation methods, stability and applications of hybrid nanofluids. *Renew Sustain Energy Rev* 80:1112–1122
- Salari A, Kazemian A, Ma T, Hakkaki-Fard A, Peng J (2020) Nanofluid based photovoltaic thermal systems integrated with phase change materials: numerical simulation and thermodynamic analysis. *Energy Convers Manag* 205:112384
- Das L, Habib K, Saidur R, Aslfattahi N, Yahya SM, Rubbi F (2020) Improved thermophysical properties and energy efficiency of aqueous ionic liquid/mxene nanofluid in a hybrid pv/t solar system. *Nanomaterials* 10(7):1–26
- Asadi M, Asadi A (2016) Dynamic viscosity of MWCNT/ZnO-engine oil hybrid nanofluid: an experimental investigation and new correlation in different temperatures and solid concentrations. *Int Commun Heat Mass Transf* 76:41–45
- Sarafraz M, Safaei MR, Leon A, Tilili I, Alkanhal T, Tian Z, Goodarzi M, Arjomandi M (2019) Experimental investigation on thermal performance of a PV/T-PCM (photovoltaic/thermal) system cooling with a PCM and nanofluid. *Energies* 12:2572
- Ekiciler R, Arslan K, Turgut O, Kurşun B (2020) Effect of hybrid nanofluid on heat transfer performance of parabolic trough solar collector receiver. *J Therm Anal Calorim* 143(2):1637–1654
- Murshed SMS, Estellé P (2017) A state of the art review on viscosity of nanofluids. *Renew Sustain Energy Rev* 76:1134–1152
- Bashimezhad K, Bazri S, Safaei MR, Goodarzi M, Dahari M, Mahian O, Dalkılıça AS, Wongwises S (2016) Viscosity of nanofluids: a review of recent experimental studies. *Int Commun Heat Mass Transf* 73:114–123
- Duangthongsuk W, Wongwises S (2009) Measurement of temperature-dependent thermal conductivity and viscosity of TiO<sub>2</sub>-water nanofluids. *Exp Therm Fluid Sci* 33(4):706–714
- Chandrasekar M, Suresh S, Chandra Bose A (2010) Experimental investigations and theoretical determination of thermal conductivity and viscosity of Al<sub>2</sub>O<sub>3</sub>/water nanofluid. *Exp Therm Fluid Sci* 34(2):210–216
- Mahbulul IM, Saidur R, Amalina MA (2012) Latest developments on the viscosity of nanofluids. *Int J Heat Mass Transf* 55(4):874–885
- Corcione M (2011) Empirical correlating equations for predicting the effective thermal conductivity and dynamic viscosity of nanofluids. *Energy Convers Manag* 52(1):789–793
- Okonkwo EC, Wole-Osho I, Almanassra IW, Abdullatif YM, Al-Ansari T (2020) An updated review of nanofluids in various heat transfer devices. *J Therm Anal Calorim* 145(6):2817–2872
- Almanassra IW, Okonkwo EC, Alhassan O, Ali M, Kochkodan V, Al-ansari T (2021) Stability and thermophysical properties test of carbide-derived carbon thermal fluid; a comparison between functionalized and emulsified suspensions. *Powder Technol* 377:415–428
- Afrand M, Nazari Najafabadi K, Akbari M (2016) Effects of temperature and solid volume fraction on viscosity of SiO<sub>2</sub>-MWCNTs/SAE40 hybrid nanofluid as a coolant and lubricant in heat engines. *Appl Therm Eng* 102:45–54
- Giwa SO, Sharifpur M, Goodarzi M, Alsulami H, Meyer JP (2021) Influence of base fluid, temperature, and concentration on the thermophysical properties of hybrid nanofluids of alumina-ferrofluid: experimental data, modeling through enhanced ANN, ANFIS, and curve fitting. *J Therm Anal Calorim* 143(6):4149–4167
- Goodarzi M, Toghraie D, Reiszadeh M, Afrand M (2019) Experimental evaluation of dynamic viscosity of ZnO-MWCNTs/engine oil hybrid nanolubricant based on changes in temperature and concentration. *J Therm Anal Calorim* 136(2):513–525
- Einstein A (1906) Eine neue Bestimmung der Molekuldimensionen. *Ann Phys* 19:289–306
- Chen H, Ding Y, Tan C (2007) Rheological behaviour of nanofluids. *New J Phys* 9:367
- Okonkwo EC, Wole-Osho I, Kavaz D, Abid M, Al-Ansari T (2020) Thermodynamic evaluation and optimization of a flat plate collector operating with alumina and iron mono and hybrid nanofluids. *Sustain Energy Technol Assess* 37:100636
- Wole-Osho I, Okonkwo EC, Kavaz D, Abbasoglu S (2020) An experimental investigation into the effect of particle mixture ratio on specific heat capacity and dynamic viscosity of Al<sub>2</sub>O<sub>3</sub>-ZnO hybrid nanofluids. *Powder Technol* 363:699–716
- Eshgarf H, Sina N, Esfe MH, Izadi F, Afrand M (2018) Prediction of rheological behavior of MWCNTs-SiO<sub>2</sub>/EG-water non-Newtonian hybrid nanofluid by designing new correlations and optimal artificial neural networks. *J Therm Anal Calorim* 132(2):1029–1038
- Alarifi IM, Nguyen HM, Bakhtiyari AN, Asadi A (2019) Feasibility of ANFIS-PSO and ANFIS-GA models in predicting thermophysical properties of Al<sub>2</sub>O<sub>3</sub>-MWCNT/Oil hybrid nanofluid. *Materials (Basel)* 12(21):3628
- Karimipour A, Bagherzadeh SA, Taghipour A, Abdollahi A, Safaei MR (2019) A novel nonlinear regression model of SVR as a substitute for ANN to predict conductivity of MWCNT-CuO/water hybrid nanofluid based on empirical data. *Phys A Stat Mech Appl* 521:89–97
- Tian Z, Arasteh H, Parsian A, Karimipour A, Safaei MR, Nguyen TK (2019) Estimate the shear rate & apparent viscosity of multiphased non-Newtonian hybrid nanofluids via new developed support vector machine method coupled with sensitivity analysis. *Phys A Stat Mech Appl* 535:122456
- Alrashed AAAA, Gharibdousti MS, Goodarzi M, de Oliveira LR, Safaei MR, Bandarra Filho EP (2018) Effects on thermophysical properties of carbon based nanofluids: Experimental data, modelling using regression, ANFIS and ANN. *Int J Heat Mass Transf* 125:920–932
- Bahrami M, Akbari M, Bagherzadeh SA, Karimipour A, Afrand M, Goodarzi M (2019) Develop 24 dissimilar ANNs by suitable architectures & training algorithms via sensitivity analysis to better statistical presentation: measure MSEs between targets &

- ANN for Fe–CuO/Eg–water nanofluid. *Phys A Stat Mech Appl* 519:159–168
28. Ahmadi MH, Mohseni-Gharyehsafa B, Ghazvini M, Goodarzi M, Jilte RD, Kumar R (2020) Comparing various machine learning approaches in modeling the dynamic viscosity of CuO/water nanofluid. *J Therm Anal Calorim* 139(4):2585–2599
  29. Boyle, P. (2007) Gaussian processes for regression and optimisation. 190
  30. Khalifeh A, Vaferi B (2019) Intelligent assessment of effect of aggregation on thermal conductivity of nanofluids—comparison by experimental data and empirical correlations. *Thermochim Acta* 681:1783
  31. Wole-Osho I, Okonkwo EC, Adun H, Kavaz D, Abbasoglu S (2020) An intelligent approach to predicting the effect of nanoparticle mixture ratio, concentration and temperature on thermal conductivity of hybrid nanofluids. *J Therm Anal Calorim* 144(3):671–688
  32. Wang X, Xu X, Choi SUS (1999) Thermal conductivity of nanoparticle-fluid mixture. *J Thermophys Heat Transf* 13:474–480
  33. Okonkwo EC, Wole-Osho I, Kavaz D, Abid M (2019) Comparison of experimental and theoretical methods of obtaining the thermal properties of alumina/iron mono and hybrid nanofluids. *J Mol Liq* 292:111377
  34. Afshari A, Akbari M, Toghraie D, Yazdi ME (2018) Experimental investigation of rheological behavior of the hybrid nanofluid of MWCNT–alumina/water (80%)–ethylene-glycol (20%): new correlation and margin of deviation. *J Therm Anal Calorim* 132(2):1001–1015
  35. Hemmat Esfe M, Reiszadeh M, Esfandeh S, Afrand M (2018) Optimization of MWCNTs (10%) – Al<sub>2</sub>O<sub>3</sub> (90%)/5W50 nanofluid viscosity using experimental data and artificial neural network. *Phys A Stat Mech Appl* 512:731–744
  36. Soltani O, Akbari M (2016) Effects of temperature and particles concentration on the dynamic viscosity of MgO-MWCNT/ethylene glycol hybrid nanofluid: experimental study. *Phys E Low-Dimens Syst Nanostruct* 84:564–570
  37. Dardan E, Afrand M, Meghdadi Isfahani AH (2016) Effect of suspending hybrid nano-additives on rheological behavior of engine oil and pumping power. *Appl Therm Eng* 109:524–534
  38. Nabil MF, Azmi WH, Abdul Hamid K, Mamat R, Hagos FY (2017) An experimental study on the thermal conductivity and dynamic viscosity of TiO<sub>2</sub>-SiO<sub>2</sub> nanofluids in water: ethylene glycol mixture. *Int Commun Heat Mass Transf* 86:181–189
  39. Ruhani B, Toghraie D, Hekmatifar M, Hadian M (2019) Statistical investigation for developing a new model for rheological behavior of ZnO–Ag (50%–50%)/Water hybrid Newtonian nanofluid using experimental data. *Phys A Stat Mech Appl* 525:741–751
  40. Hemmat Esfe M, Afrand M, Yan WM, Yarmand H, Toghraie D, Dahari M (2016) Effects of temperature and concentration on rheological behavior of MWCNTs/SiO<sub>2</sub>(20–80)-SAE40 hybrid nano-lubricant. *Int Commun Heat Mass Transf* 76:133–138
  41. Hemmat Esfe M, Rostamian H, Reza Sarlak M (2018) A novel study on rheological behavior of ZnO-MWCNT/10w40 nanofluid for automotive engines. *J Mol Liq* 254:406–413
  42. Afrand M, Nazari Najafabadi K, Sina N, Safaei MR, Kherbeet AS, Wongwises S, Dahari M (2016) Prediction of dynamic viscosity of a hybrid nano-lubricant by an optimal artificial neural network. *Int Commun Heat Mass Transf* 76:209–214
  43. Hemmat Esfe M, Afrand M, Rostamian SH, Toghraie D (2017) Examination of rheological behavior of MWCNTs/ZnO-*SAE40* hybrid nano-lubricants under various temperatures and solid volume fractions. *Exp Therm Fluid Sci* 80:384–390
  44. Ramezanizadeh M, Ahmadi MH, Nazari MA, Sadeghzadeh M, Chen L (2019) A review on the utilized machine learning approaches for modeling the dynamic viscosity of nanofluids. *Renew Sustain Energy Rev* 114:109345
  45. Hamid KA, Azmi WH, Nabil MF, Mamat R, Sharma KV (2018) Experimental investigation of thermal conductivity and dynamic viscosity on nanoparticle mixture ratios of TiO<sub>2</sub>-SiO<sub>2</sub> nanofluids. *Int J Heat Mass Transf* 116:1143–1152
  46. Wanatasanapan VV, Abdullah MZ, Gunnasegaran P (2020) Effect of TiO<sub>2</sub>-Al<sub>2</sub>O<sub>3</sub>nanoparticle mixing ratio on the thermal conductivity, rheological properties, and dynamic viscosity of water-based hybrid nanofluid. *J Mater Res Technol* 9(6):13781–13792
  47. Wole-Osho I, Adun H, Adedeji M, Okonkwo EC, Kavaz D, Dagbasi M (2020) Effect of hybrid nanofluids mixture ratio on the performance of a photovoltaic thermal collector. *Int J Energy Res* 44(11):9064–9081
  48. Mohammed A, Burhan L, Ghafor K, Sarwar W, Mahmood W (2021) Artificial neural network (ANN), M5P-tree, and regression analyses to predict the early age compression strength of concrete modified with DBC-21 and VK-98 polymers. *Neural Comput Appl* 33(13):7851–7873
  49. Jamei M, Ahmadianfar I, Olumegbon IA, Karbasi M, Asadi A (2021) On the assessment of specific heat capacity of nanofluids for solar energy applications: application of Gaussian process regression (GPR) approach. *J Energy Storage* 33:102067
  50. Cao C, Liao J, Hou Z, Wang G, Feng W, Fang Y (2020) Parametric uncertainty analysis for CO<sub>2</sub> sequestration based on distance correlation and support vector regression. *J Nat Gas Sci Eng* 77:103237
  51. Daneshfar R, Bemani A, Hadipoor M, Sharifpur M, Ali HM, Mahariq I, Abdeljawad T (2020) Estimating the heat capacity of non-Newtonian ionanofluid systems using ANN, ANFIS, and SGB tree algorithms. *Appl Sci* 10(18):6432
  52. Wu H, Bagherzadeh SA, D’Orazio A, Habibollahi N, Karimipour A, Goodarzi M, Bach QV (2019) Present a new multi objective optimization statistical Pareto frontier method composed of artificial neural network and multi objective genetic algorithm to improve the pipe flow hydrodynamic and thermal properties such as pressure drop and heat transfer. *Phys A Stat Mech Appl* 535:122409
  53. Cheng K, Lu Z, Zhou Y, Shi Y, Wei Y (2017) Global sensitivity analysis using support vector regression. *Appl Math Model* 49:587–598
  54. Alade IO, Rahman MAA, Saleh TA (2020) An approach to predict the isobaric specific heat capacity of nitrides/ethylene glycol-based nanofluids using support vector regression. *J. Energy Storage* 29:101313
  55. Chen H, Ding Y, Tan C (2007) Rheological behaviour of nanofluids. *New J Phys* 9(10):367
  56. Meybodi MK, Naseri S, Shokrollahi A, Daryasafar A (2015) Prediction of viscosity of water-based Al<sub>2</sub>O<sub>3</sub>, TiO<sub>2</sub>, SiO<sub>2</sub>, and CuO nanofluids using a reliable approach. *Chemom Intell Lab Syst* 149:60–69
  57. Friedman JH, Meulman JJ (2003) Multiple additive regression trees with application in epidemiology. *Stat Med* 22(9):1365–1381
  58. Landwehr N, Hall M, Frank E (2005) Logistic model trees. *Mach Learn* 59(1–2):161–205
  59. Hemmat Esfe M, Abbasian Arani AA, Rezaie M, Yan WM, Karimipour A (2015) Experimental determination of thermal conductivity and dynamic viscosity of Ag-MgO/water hybrid nanofluid. *Int Commun Heat Mass Transf* 66:189–195
  60. Yan, S.R., Kalbasi, R., Nguyen, Q., and Karimipour, A. (2020) Rheological behavior of hybrid MWCNTs-TiO<sub>2</sub>/EG nanofluid: A comprehensive modeling and experimental study. *J. Mol. Liq.*, **308**, 113058.

61. Brinkman HC (1952) The viscosity of concentrated suspensions and solutions. *J Chem Phys* 20(4):571
62. Lundgren T (1972) Slow flow through stationary random beds and suspensions of spheres. *J Fluid Mech* 51:273–299
63. Li L, Zhai Y, Jin Y, Wang J, Wang H, Ma M (2020) Stability, thermal performance and artificial neural network modeling of viscosity and thermal conductivity of Al<sub>2</sub>O<sub>3</sub>-ethylene glycol nanofluids. *Powder Technol* 363:360–368
64. Ma M, Zhai Y, Wang J, Yao P, Wang H (2020) Statistical image analysis of uniformity of hybrid nanofluids and prediction models of thermophysical parameters based on artificial neural network (ANN). *Powder Technol* 362:257–266
65. Hemmati-Sarapardeh A, Varamesh A, Husein MM, Karan K (2017) (2018) On the evaluation of the viscosity of nanofluid systems: modeling and data assessment. *Renew Sustain Energy Rev* 81:313–329
66. Gholizadeh M, Jamei M, Ahmadianfar I, Pourrajab R (2020) Prediction of nanofluids viscosity using random forest (RF) approach. *Chemom Intell Lab Syst* 201:104010
67. Jamei M, Ahmadianfar I (2020) A rigorous model for prediction of viscosity of oil-based hybrid nanofluids. *Physica A* 556:124827
68. Doan CHID, Liong S (2004) Generalization for multilayer neural network Bayesian regularization or early stopping. *Network* 119260:1–8
69. Alade IO, Abd Rahman MA, Saleh TA (2019) Modeling and prediction of the specific heat capacity of Al<sub>2</sub>O<sub>3</sub>/water nanofluids using hybrid genetic algorithm/support vector regression model. *Nano-Struct Nano-Objects* 17:103–111
70. Elith J, Leathwick JR, Hastie T (2008) A working guide to boosted regression trees. *J Anim Ecol* 77(4):802–813
71. Heidari E, Sobati MA, Movahedirad S (2016) Accurate prediction of nanofluid viscosity using a multilayer perceptron artificial neural network (MLP-ANN). *Chemom Intell Lab Syst* 155:73–85
72. Kayri M (2016) Predictive abilities of Bayesian regularization and Levenberg-Marquardt algorithms in artificial neural networks: a comparative empirical study on social data. *Math Comput Appl* 21(2):20
73. Chaipimonplin T (2016) The efficiency of using different of learning algorithms in artificial neural network model for flood forecasting at Upper River Ping catchment. *J Civi Eng* 20(1):478–484
74. Karimi H, Yousefi F, Rahimi MR (2011) Correlation of viscosity in nanofluids using genetic algorithm-neural network (GA-NN). *Heat Mass Transf und Stoffuebertragung* 47(11):1417–1425

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

## Authors and Affiliations

Humphrey Adun<sup>1</sup> · Ifeoluwa Wole-Osho<sup>1</sup> · Eric C. Okonkwo<sup>2</sup> · Tonderai Ruwa<sup>1</sup> · Terfa Agwa<sup>3</sup> · Kenechi Onochie<sup>3</sup> · Henry Ukwu<sup>4</sup> · Olusola Bamisile<sup>5</sup> · Mustafa Dagbasi<sup>1</sup>

✉ Humphrey Adun  
hadun@ciu.edu.tr

<sup>1</sup> Energy Systems Engineering Department, Cyprus International University, Mersin 10, Haspolat-Lefkosa, Turkey

<sup>2</sup> Division of Sustainable Development, College of Science and Engineering, Hamad Bin Khalifa University, Qatar Foundation, Education City, Doha, Qatar

<sup>3</sup> Civil Engineering Department, Cyprus International University, Mersin 10, Haspolat-Lefkosa, Turkey

<sup>4</sup> Computer Engineering Department, Cyprus International University, Haspolat-Lefkosa, Mersin 10, Turkey

<sup>5</sup> School of Mechanical and Electrical Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan, People's Republic of China