Case Studies in Chemical and Environmental Engineering 11 (2025) 101180

Contents lists available at ScienceDirect

Case Studies in Chemical and Environmental Engineering

journal homepage: www.sciencedirect.com/journal/case-studies-in-chemical-and-environmental-engineering

Case Report

Prediction of the viscosity of iron-CuO/water-ethylene glycol non-Newtonian hybrid nanofluids using different machine learning algorithms

Mohammed Shorbaz Graish [a], Ali B.M. Ali [b], Murtadha M. Al-Zahiwat [c], Saja Mohsen Alardhi [d], Mohammadreza Baghoolizadeh [e,**], Soheil Salahshour [f,g,h], Mostafa Pirmoradian [i,*]

[a] Department of Chemical Engineering, University of Technology- Iraq, Baghdad, 10066, Iraq
[b] Air Conditioning Engineering Department, College of Engineering, University of Warith Al-Anbiyaa, Karbala, Iraq
[c] Department of Chemical Engineering, College of Engineering, University of Misan, Amarah, Iraq
[d] Nanotechnology and Advanced Materials Research Center, University of Technology – Iraq, Baghdad, Iraq
[e] Department of Mechanical Engineering, Shahrekord University, Shahrekord, 88186-34141, Iran
[f] Faculty of Engineering and Natural Sciences, Istanbul Okan University, Istanbul, Turkey
[g] Faculty of Engineering and Natural Sciences, Bahcesehir University, Istanbul, Turkey
[h] Research Center of Applied Mathematics, Khazar University, Baku, Azerbaijan
[i] Department of Mechanical Engineering, Khomeinishahr branch, Islamic Azad University, Khomeinishahr, Iran

ARTICLE INFO

Keywords:
Machine learning algorithms
Non-Newtonian hybrid nano- antifreeze
Viscosity

ABSTRACT

Viscosity is a crucial parameter for heat transfer systems, governing pumping power, Rayleigh number, and Reynolds number; thus, viscosity prediction for hybrid nanofluids is important. Although some studies have employed ML algorithms for predicting viscosity, limited ML algorithms or specific nanofluid types were examined in previous studies, disregarding the complexities involved in the rheological behavior of a complex nanofluid system such as non-Newtonian hybrid nanofluids. To overcome this limitation, this study offers a practical contribution by utilizing 20 different machine-learning models to predict the viscosity of iron-CuO/water-ethylene glycol non-Newtonian hybrid nanofluids. The influences of the input variables: solid volume fraction (SVF), temperature, and shear rate on viscosity prediction are systematically assessed. We evaluate the prediction accuracy and reliability of algorithms using ten performance metrics including RMSE, MAE, R^2 and NSE. Multivariate Polynomial Regression (MPR) outperforms the other algorithms, which is evident in the highest correlation coefficient ($R^2 = 0.992$) and lowest error metrics. At the other end, is the Extreme Learning Machine (ELM), which turns out to be the worst performer. A unique contribution of this paper is that we extract a mathematical equation from the MPR model that allows for straightforward calculation of viscosity, avoiding non-trivial ML computations. This simplicity aids in practical applications and increases usefulness for engineers and researchers alike. Using advanced data visualization techniques (heatmaps, box plots, KDE plots and Taylor diagrams), the relationships between input variables and viscosity as well as the model performance are explored. These results give a better understanding of the non-Newtonian hybrid nanofluid behavior and a solid predictor of design-efficient heat transfer systems.

* Corresponding author.
** Corresponding author.
E-mail addresses: mohamadbaghoolizadeh@gmail.com (M. Baghoolizadeh), m.pirmoradian@iaukhsh.ac.ir, mostafaa.pirmoradian@gmail.com (M. Pirmoradian).

https://doi.org/10.1016/j.cscee.2025.101180
Received 12 February 2025; Received in revised form 2 March 2025; Accepted 4 March 2025
Available online 7 March 2025

Nomenclature

Solid volume fraction (SVF)	φ (%)	Viscosity	μ_{nf} (cP)
Temperature	T (°C)	Shear rate	γ (RPM)
Machine Learning Algorithm	MLA	Decision Tree	DT
Multi-Layer Perceptron	MLP	Adaptive Neuro-Fuzzy Inference System	ANFIS
Gradient Descent	GD	Group Method of Data Handling	GMDH
Radial Basis Function	RBF	Back Propagation Neural Network	BPNN
BFGS Quasi-Newton	BFGS	Levenberg-Marquardt	LM
Support Vector Machine	SVM	Bayesian network	BN
Extreme Learning Machine	ELM	Elastic Component Regression	ECR
Multiple Linear Regression	MLR	Gaussian Process Regression	GPR
XGBoost	XGB	Multivariate Polynomial Regression	MPR
Partial Least Squares Regression	PLR	Least Absolute Shrinkage and Selection Operator	Lasso
Evaluation Criteria	EC	Analysis Plots	AP
Mean Absolute Relative Error	MARE	Root Mean Squared Error	RMSE
Nash–Sutcliffe Model Efficiency Coefficient	NSE	Correlation Coefficient	R
Mean Squared Error	MSE	Mean Absolute Percentage Error	MAPE
R squared	R^2	Kernel density estimate	Kde
Standard Deviation	STD	Root Mean Squared Deviation	RMSD
Mean Absolute Error	MAE	Mean Bias Error	MBE
Coefficient of the Variation of the Root Mean Square Error	CvRMSE		

1. Introduction

Heat exchanger performance is a function of a number of variables including heat transfer coefficient, cross-sectional area, and temperature difference. Water, ethylene glycol, and oils are only some of the working fluids being utilized in industry and engineering. The main problem with these fluids is that they have poor heat transfer and low thermal conductivity. One of the new methods that have been utilized in order to increase the TC of such fluids is suspending nanoparticles (NPs) with enhanced thermal properties in the working fluid to create a material called nanofluid (NF). Choi [1] coined the term "nanofluid" to refer to a fluid that suspends extremely small particles (NPs with a diameter smaller than 100 nm). NFs involve the addition of one or more solid phases to a fluid, resulting in an enhanced heat transfer rate and altered viscosity. Certain NPs, like aluminum oxide and magnesium oxide, exist as metal oxides and can be readily distributed and suspended in liquids. Diamond, carbon nanotubes, and other materials will greatly enhance heat transfer in comparison to oxide NPs. Hence, the simultaneous use of these nanomaterials can result in a steadfast amalgamation with distinct and coveted thermal characteristics. The heat transfer coefficient is significantly influenced by viscosity, which is a key factor in the relationships that govern heat transfer. The incorporation of nanoparticles into base fluids significantly alters the thermophysical properties of nanofluids, particularly their viscosity [2–4]. It has been shown that viscosity plays a crucial role in determining pumping power, Rayleigh number, Reynolds number, and consequently, heat transfer. Numerous experiments and studies have been conducted to investigate the viscosity and heat transfer in NFs. Bashirnezhad et al. [5], Koca et al. [6], and Murshed and Estellé [7] highlighted that the viscosity of NFs is influenced by factors such as temperature, NP type, solid volume fraction (SVF), particle size, and base fluid (BF) type. While investigating heat transfer, Selvarajoo et al. [8] experimented with different volume concentrations of mono NFs Al_2O_3/water and GO/water and hybrid nanofluid (HNF) Al_2O_3-GO/water to evaluate their thermophysical properties. Viscosity and TC were examined in the temperature range of 30–50 °C. An analytical regression model was used to predict the thermal conductivity and dynamic viscosity of an Al_2O_3-GO HNF. The TC improvement of the HNF was around 4.30 % higher than that of pure Al_2O_3 and 4.34 % higher than GO mono NF. Khouri et al. [9] investigated the heat transfer properties of water-based graphene oxide (GO) nanofluids in a heat exchanger. They analyzed the nanofluid's thermal conductivity, specific heat capacity, and viscosity at different temperatures and GO concentrations. The TC increased with both temperature and NP concentration, enhancing heat transfer through conduction and convection. However, the specific heat capacity increased with temperature but decreased with higher GO concentrations. Additionally, the viscosity of the nanofluid rose with higher NP concentrations. Ajeena et al. [10] investigated the dynamic viscosity of an HNF with ZrO_2–SiC (50 %–50 %) and distilled water. To do so, a two-step process was used to disperse NPs in the BF. The viscosity of SVFs in the range of 0.025 %–0.1 % was measured at temperatures from 20 to 60 °C. The study exhibited a relationship between NF viscosity and temperature, as well as SVF parameters. The results showed that there was an increase in the dynamic viscosity value at greater solid concentrations and lower temperatures. The dynamic viscosity of NF at 20 and 60 °C was measured and showed an increase in viscosity by 29.6 % and 64.2 %, respectively, with 0.025 % NPs. This indicated that NP viscosity was more sensitive at higher temperatures. The experiments also indicated that the ZrO_2–SiC HNF was Newtonian across a range of temperatures. The research also suggested a new correlation through which the dynamic viscosity of HNF could be calculated based on the experimental parameters (temperature and SVF) with a 98.92 % error. Sepehrnia et al. [11] investigated the rheological efficiency and dynamic viscosity of an HNF made of SiO_2 and MWCNTs NPs (90:10) with 5W30 engine oil as BF at various SRs (50–1000 rpm) experimentally. The SVFs and temperatures were assessed within the ranges of 0.05–1.00 vol% and 5–65 °C, respectively. The HNF exhibited characteristics of a non-Newtonian fluid. Furthermore, the HNF exhibited pseudoplastic characteristics in all SVFs and temperatures, as the calculated power law index was less than unity. It was noted that the dynamic viscosity decreased as the NF's temperature increased, whereas the dynamic viscosity increased as the NF's SVF increased. A three-variable association was found. Dynamic viscosity sensitivity to temperature, SVF, and SR was also assessed. With a constant SR of 800 rpm, dynamic viscosity sensitivity increased with NF's temperature and concentration. Wanatasanappan et al. [12] conducted experimental investigations on the rheological and viscosity characteristics of an Al_2O_3–Fe_2O_3 HNF. After determining a correlation for viscosity prediction, they investigated how the Al_2O_3–Fe_2O_3 mixture ratio affected the viscosity property. The BF was a 60/40 mixture of water and ethylene glycol. Five distinct Al_2O_3–Fe_2O_3 NP compositions were examined at temperatures ranging from 0 to 100 °C. The 40/60 Al_2O_3–Fe_2O_3 composition exhibited the highest viscosity value at all temperatures examined, according to the experimental data. Conversely, the 60/40 composition exhibited the lowest viscosity value. Additionally, the viscosity decreased by 87.2 % as the temperature was raised from 0 to 100 °C. Furthermore, the Newtonian characteristic was observed in all Al_2O_3–Fe_2O_3 compositions. Sepehrnia et al. [13] examined the thermophysical and rheological properties of hydraulic oil HLP 68 as the BF with a ternary combination of Fe_3O_4, TiO_2, and GO nano-additives in a variety of nanomaterial mixing ratios (MRs) (1:1:1, 2:1:1, 1:2:1, and 1:1:2), SVFs (0–1 %), and temperatures (15–65°). Their analysis of all MRs showed Newtonian THNF behavior. At 15 °C, the BF viscosity increased by 345 %, 1821 %, 1763 %, and 1990 % for MRs of 1:1:1, 1:2:1, and 2:1:1, respectively, when a 1 % SVF of GO: Fe_3O_4: TiO_2 was present. In addition, at the highest SVF, the MRs of 1:1:1, 1:2:2, and 3:1:1 showed a 66 %, 75 %, 60 %, and 70 % rise in viscosity with a temperature decrease from 65 °C to 15 °C, respectively. The experimental, statistical, and numerical viscosity of a MWCNT (50 %)-MgO (50 %)/SAE40 HNF was measured by Esfe et al. [14]. Temperatures, SVFs, and shear rates were assessed within the ranges of 25–50 °C, 0.0625–1 %, and 666.5–9331 s^{-1}, respectively. According to their

findings, the HNF exhibited properties of a non-Newtonian fluid that was pseudo-plastic. At a specific amount of SVF (0.0625 %), temperature (25 °C), and specific rate (3999 s^{-1}), the maximum viscosity decrease was 5.91 % and the maximum viscosity rise was 28.70 %. In another study by Ajeena et al. [15], the viscosity of ZrO_2/DW and SiC/DW NFs was measured over a temperature range of 20–60 °C with various SVFs (0.025, 0.05, 0.075, and 0.1 %). The dynamic viscosity of the sample was enhanced by increasing the SVF of the NPs while lowering the temperature. Consequently, NPs had a more noticeable effect on the viscosity as the temperature increased. The results demonstrated that the viscosity increase for the ZrO_2/DW NF topped at 226.3 %, while for the SiC/DW NF, it was 110.5 %. Thermophysical properties of MWCNT-ZnO (30–70 %)/SAE40 HNF were studied by Esfe et al. [16] over a variety of SVFs, temperatures, and SRs. At T = 31.156 °C, SVF = 0.063 %, and SR = 933.923 s^{-1}, their results demonstrated that the ideal viscosity value was 209.53 mPa.sec. At temperatures ranging from 10 to 50 °C, with five SVFs of magnetocaloric NF and varying SRs, Abbasian et al. [17] investigated the rheological properties of $CoFe_2O_4$ superparamagnetic NPs dispersed in water-ethylene glycol (EG) coolant. Dispersed in a 50:50 mixture of EG and water, the cobalt ferrite metallic complexes were prepared by solvothermal methods. The experimental results of the magnetocaloric NF demonstrated non-Newtonian dynamics. An increase in NP mass concentration from 0.05 % to 0.8 % increased viscosity by almost 80 % at 10 °C. Furthermore, TC was enhanced with increasing temperature, reaching a high of 9.4 % at 50 °C. In their study, Hafeez et al. [18] investigated the effects of Cu–Al_2O_3 NPs on the rheological properties, dynamic viscosity, and TC of HNFs based on kerosene oil. For numerical analysis, a mathematical model of an HNF was derived from PDEs, which were then transformed into ODEs using similarity conversion. Both the friction coefficient and the heat transfer rate were shown to be enhanced by the introduction of NPs. Esfe et al. [19] used different RSM models to figure out the dynamic viscosity of a SiO_2-MWCNT HNF in SAE40 oil, where 60 % SiO_2 and 40 % MWCNT made up the HNF. The study was performed under the following conditions: temperature = 25–50 °C, SVF = 0–1% and SR = 666.5–9331 s^{-1}. The viscosity of HNFs was determined using the correlation functions of several models. The impact of various parameters on HNF viscosity was examined. SR and temperature had the least and most significant effects, respectively.

While molecular dynamics simulations and laboratory investigations have yielded efficient and practical insights into NF characteristics, the time and money required to run these models have prompted academics to seek other modeling approaches. Fuzzy logic, adaptive neuro-fuzzy inference system (ANFIS), genetic algorithm (GA), and artificial neural network (ANN) are some of the methods classified under this category. Their adaptability, superior effectiveness, and accuracy have been demonstrated in several scientific domains [20–23]. The described predictive models are self-improving and data-driven; thus, they can anticipate process outcomes with high accuracy. They also have great correlation results and can model nonlinear relationships with a variety of input variables. Concurrently, it saves time and money by not requiring any more experiments. In this regard, Esfe et al. [23] carried out an investigation in which they employed an ANN to predict the viscosity (μ_{nf}) of an NF composed of MWCNT-MgO/SAE40 engine oil at different temperatures, SVF, and SR. The Levenberg-Marquardt (ML) learning algorithm was employed in an MLP ANN consisting of two hidden layers. The first layer contained the optimum structure of 10 neurons, while the second layer contained 4 neurons. Concentration, SR, and temperature were used as input parameters for ANN modeling, while the predicted μ_{nf} was observed as the output parameter. The optimum ANN, as per the data, possessed the least mean square error and consisted of eight neurons in both of the layers. As per their observation, ANN prediction of data was much better compared to correlation. By applying a mix of heuristic techniques and an ANFIS, Wang et al. [24] identified the optimal temperature and SVF pair for an HNF made of silicon oxide, MWCNT, aluminum, and water with the highest TC (k_{nf}) and the lowest dynamic viscosity. Four elite algorithms—strong Pareto evolutionary algorithm II, Pareto envelope-based selection algorithm II, non-dominated sorting GA II, and multi-objective particle swarm optimization—were applied in the selection of the most significant set of input parameters. Zhang et al. [25] proposed a hybrid method combining ML, MOO, and MCDM to select the optimum parameters of MWCNTs-oxide HNFs based on water. They used two effective ML methods, the group method of data handling neural network (GMDH-NN) and the combinatorial (COMBI) algorithm, to simulate four important TPPs: density ratio, viscosity ratio, specific heat capacity ratio, and TC ratio. The optimization variables that were considered were the type of oxide NP, SVF, and system temperature. Different types of oxide-NPs showed substantially different distributions of optimum points over various temperature ranges. To model the relationship between the dynamic viscosity of the MgO-SAE 5W30 Oil HNF and three critical parameters, namely the SVF, temperature, and SR, Gao et al. [26] employed an RBF-ANN. Their findings demonstrated that the dynamic viscosity of this NF was reduced as the temperature and SR were increased. Contrarily, although this result can be ignored, the production was directly affected by the volume proportion of NPs. A drop in dynamic viscosity would result from a temperature increase of 5–55 °C. The dynamic viscosity dropped from 400 cP to 25 cP when the SR was increased from 50 rpm to 1000 rpm. In their study, Fadhl et al. [27] modeled the dynamic viscosity of NFs containing MgO NPs using two intelligence approaches: ANFIS and GMDH. Their findings indicated that GMDH had better precision. Utilizing various machine learning models including SVR, ANN, and ANFIS, AbuShanab et al. [28] aimed to forecast the dynamic viscosity of NFs, namely Polyalpha-Olefin-hexagonal boron nitride. There were 540 points of experimental data used to train and evaluate the models. While all three models were able to correctly estimate the viscosity of NFs, the ANFIS and ANN models performed better than the SVR model. Although both the ANFIS and ANN models performed similarly, the ANN model was chosen because it trained and computed faster. As the SR parameter was removed from the input layer, the ANN model became more accurate. Their research proved that ANN and other machine learning models could accurately predict the dynamic viscosity of the NF. Hua et al. [29] evaluated the dynamic viscosity of a hybrid antifreeze with MWCNTs, aluminum oxide, and water and ethylene-glycol in response to SVF and temperature. The NF's anticipated viscosity was determined using an ANN trained on samples at temperatures of 25–50 °C, and SVFs of 0.25–1 %. The NF's viscosity was impacted in various ways by SR and SVF. While rising SR greatly lowered viscosity mean and variance, increasing SVF increased deviation and mean values. The change in viscosity due to temperature was smaller compared to the variations caused by SR and SVF. The two-layer network and thirteen neurons with nonlinear activation functions in the hidden layer of the suggested ANN model predicted viscosity versus inputs accurately. To improve the dynamic viscosity of an HNF composed of MWCNT-Al_2O_3 (40:60)/Oil 5W50, Esfe et al. [30] employed ANN. To determine which parameters of MWCNT-Al_2O_3 (40:60)/Oil 5W50 HNF were most important for dynamic viscosity, a sensitivity analysis was performed. The results showed that dynamic viscosity values were highest at temperatures lower than 5 °C. Changes in SR between zero and eight hundred revolutions per minute also decreased the dynamic viscosity. Dai et al. [31] used ANN to evaluate the effects of SVF, SR, and temperature on the dynamic viscosity and torque of SiO_2/EG NF. Rheological properties were predicted using many machine learning (ML) models, and the best model was picked. All samples showed that torque increased linearly with SR, steeper at lower temperatures. Gaussian Process Regression (GPR) models with the Matérn covariance function predicted dynamic viscosity well on both datasets. Singh and Ghosh [32] suggested a single feed-forward MLPNN to predict the density and dynamic viscosity of graphene NP/DW, aluminum oxide/DW, and MWCNT NFs over 30–80 °C. Dynamic viscosity and density were output from temperature, volume concentration, and NFs that were input. As NF concentration

grew, density and viscosity decreased, and vice versa as temperature climbed. Experimental findings were compared to MLPNN and mathematical model results. The greatest viscosity and density errors were less than 1 % and 0.2 %, respectively. In another study, Said et al. [33] generated ethylene glycol (EG) NFs utilizing rGO-Fe_3O_4–TiO_2 ternary hybrid nanocomposites. To measure density and viscosity, they varied temperatures from 25 to 50 °C and weight fraction from 0.01 to 0.25 %. Modern machine learning techniques including ANN, BRT, and SVM were used to build the forecasting model. It was found that SVM, ANN, and BRT could accurately duplicate lab density and viscosity data. Kanti et al. [34] research examined the effects of concentration and temperature on the viscosity and thermal conductivity of graphene oxide, silicon dioxide, and titanium dioxide water-based nanofluids. Thermal conductivity and viscosity were found to be higher in the nanofluids compared to water, with graphene oxide recording the best result. Deep learning, deep neural networks, and a high gradient amplification approach were utilized in the study to handle complex data properly. The extreme slope model of the viscosity model had a lower R^2 value (0.9122) and mean squared error (0.010) than the deep neural network-based model (0.3329). In another study, Kanti et al. [35] investigated silicon dioxide, graphene oxide, titanium dioxide, and hybrid water-based nanofluids and observed elevated viscosity and thermal conductivity, in addition to improved performance. Optimum thermal conductivity and viscosity ratio for graphene oxide nanofluids were reported at 60 °C and 30 °C, respectively, with an enhancement of 52 and 177 % relative to the base fluid. Graphene oxide-TiO_2 hybrid nanofluids exhibited thermal conductivity and viscosity ratios of 43 % and 144 % higher than the base fluid under the same conditions. Thermophysical properties of hybrid nanofluids were predicted by state-of-the-art machine learning models, with the random forest model being the most versatile. Nanofluid concentration played a significant role in the prediction of the TC ratio but not the viscosity ratio. Akilu et al. [36] investigated the heat transfer behavior and friction factor of silicon dioxide and glycerol-based ethylene glycol nanofluids, which were higher than the base fluid. It was observed from the study that ethylene glycol-based glycerol and silicon dioxide nanofluid exhibited maximum heat transfer improvement by 5.4 % and 8.3 %, respectively. The research employed five machine learning methods for the prediction of intricate experimental data, which were linear regression, random forest, steep gradient boosting, adaptive boosting, and decision trees. The models successfully predicted nanofluid thermal performance, which played a part in the development of efficient cooling systems and the stability of energy systems. Sharma et al. [37] explored the thermophysical characteristics of polydisperse SiO_2 nanoparticles in a glycerol-water mixture. A two-step method was used to prepare aqueous glycerol nanofluid containing 30 % glycerol (30 GW) and SiO_2 with particle sizes of 15, 50, and 100 nm. The study measured the properties of nanofluids at 30–100 °C, revealing a stable single-phase liquid, decreased viscosity and density, increased thermal conductivity, and slightly increased specific heat. At 60 °C, the thermal conductivity and viscosity of a 0.5 % SiO_2 solution changed by 11.1 % and 32 %, respectively, compared to the base liquid. The Gaussian process regression optimization Bayesian approach, an explainable artificial intelligence technique, was used to develop a predictive model for nanofluid properties, enhancing predictability and explainability through kernel functions and historical data representation, resulting in high correlation values and minimal modeling errors.

The present study introduces a novel and comprehensive approach to predict the viscosity of iron-CuO/water-Ethylene glycol non-Newtonian hybrid nanofluids, employing a diverse array of machine learning algorithms. The innovation resides in the systematic and meticulous evaluation of twenty distinct machine learning algorithms, each with unique strengths and applications, to identify the most optimal model for viscosity prediction. This approach not only enhances prediction accuracy but also elucidates the complex interactions between input variables (SVF, temperature, and shear rate) and the output variable (viscosity). The comprehension of input variable behavior through advanced data analysis techniques such as heat maps and box plots is a requirement, and this study contributes greatly to such an important relevance. These techniques ease feature selection and ensure that useful data are used to train ML models that could enhance the latter's predictive ability. The critical result obtained from this study is the mathematical formula describing the relationship between the dynamic viscosity of hybrid nanofluids and the following parameters; temperature, shear rate, and SVF. All this was made possible by thorough analysis of the employed experimental data along with graphical tools like error histograms, KDE plots, and Taylor diagrams, and sophisticated error assessment methods like RMSE, MAE, and R. With these instruments, the precision, reliability, and error distribution from optimization algorithms to machine learning models can be assessed.

- In this instance, work scrutinizes, in-depth, twenty different machine learning algorithms to perhaps more accurately forecast the viscosity of a non-Newtonian hybrid nanofluid. Few works manage such painstaking comparisons, since most discussions focus only on a limited set of algorithms. The research recommends valuable insights into the best algorithms for viscosity prediction of the nanofluids; this is achieved by investigation across a broad range of alternatives.
- The study also employs heatmaps, box plots, error histograms, KDE plots, and Taylor diagrams for enhanced data analysis and interpretation. Understanding how the input factors interact also allows for optimizing nanofluids' performance in real applications.
- This study presents a new approach to formulate a relationship between viscosity and its input parameters (SVF, temperature, and shear rate). It also acts as an ancillary benchmark, in that it can now provide viscosity values without the burden of applying complex machine learning models.
- This research goes beyond basic accuracy measurements to provide a more detailed picture of how well each machine learning algorithm performed by introducing ten assessment variables.

The majority of previous studies have trained a few machine learning algorithms (ANN, SVM, or ANFIS) to predict viscosity. On the other side, this paper assesses 20 unique machine learning models for the first time and offers a complete benchmark for approximate viscosity estimation. In previous research, this kind of comprehensive comparison has never before been conducted, leading to more insights into how algorithms perform on different metrics. In previous works, models were typically evaluated using a small number of performance metrics, such as RMSE or R^2. Singh and Ghosh, for example, focused on using RMSE and MAPE. However, 10 evaluation criteria include not only RMSE, MAE, NSE, MBE, and CvRMSE to comprehensively analyze the accuracy and reliability of the models ensuring a deeper examination of the strengths and weaknesses of each algorithm. While some research studies have suggested correlations or models for viscosity prediction, most do not provide an explicit mathematical equation derived from the machine learning outputs. The mathematical equation obtained in this study by employing the MPR algorithm allows us to easily compute the viscosity without the need for endless complex computation which the ML approaches require; thus greatly improving the practical implementation status of this algorithm, and making it very useable for engineers and researchers. Most of the existing studies are well-trained in predicting the viscosity without examining the link between the input and output variables like SVF, temperature, and shear rate and viscosity. In this regard, the current study utilizes advanced data visualization techniques such as heatmaps, box plots, error histograms, KDE plots and Taylor diagrams to illustrate the influence of input variables on viscosity, and ultimately offer a more insightful perspective regarding the underlying physical phenomena governing nanofluid rheology. Even though many studies have been conducted for predicting the viscosity of mono-nanofluids or Newtonian fluid, the number of researches is limited for predicting viscosity in non-Newtonian hybrid

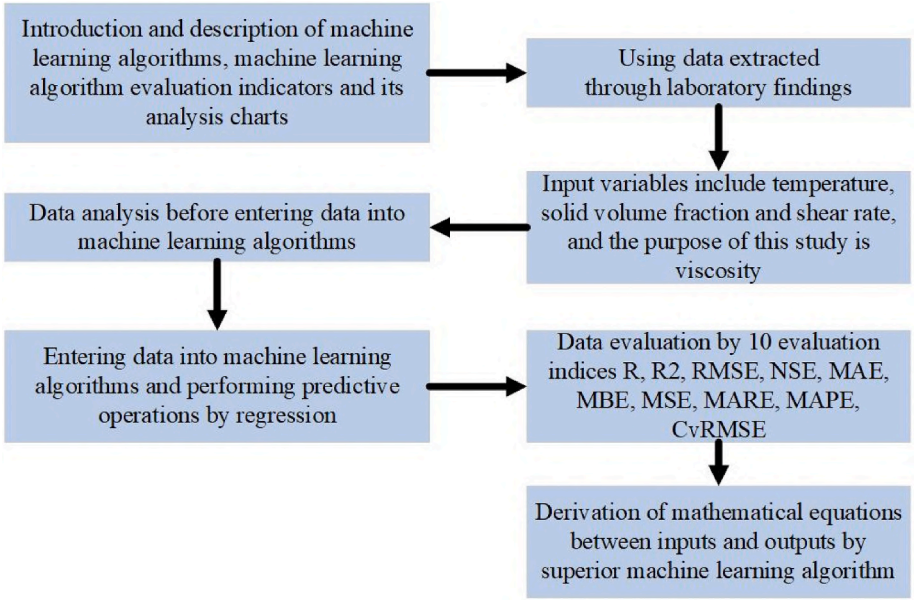


Fig. 1. The schematic of the stages of this study.

nanofluids. Despite a significant amount of research on hybrid nanofluids, this particular study focuses on a less-explored area in the literature, iron-CuO/water-ethylene glycol non-Newtonian hybrid nanofluids, which will provide further insight into their complex rheological behaviors as evidenced by the results obtained. Some research is limited to theoretical modeling and not used in real-world applications. Practical applications of the machine learning regression (MPR) model in designing heat exchangers, cooling systems, etc. are reported in this study. Industrial relevance is further augmented by the provision of an easy prediction methodology that could be implemented in real-time monitoring systems. ML approaches are also not transparent, and so their results can be difficult to interpret. Furthermore, this study employs explainable AI methods like Taylor diagrams and mathematical models to ensure clarity and interpretability, in line with contemporary trends in machine learning research, where explainability is becoming a vital concern. Our study thus makes several contributions that set it apart from previous work: extensive analysis of 20 machine learning algorithms, thorough model comparison based on ten performance metrics, derivation of a mathematical equation for predicting viscosity, utilization of advanced data analysis techniques to investigate variable interactions and the consideration of practical applications and real-world usability of prediction results through a focus on nanofluids with strong non-Newtonian properties and hybrid additives as well as explainable AI methods for greater transparency in the trained models. Overall, this enhances the novelty of the study, as well as emphasizes the importance of this work to the existing literature on nanofluid viscosity prediction.

2. Methodology

Nanofluids have been used extensively in many contexts and settings, greatly advancing both commercial operations and scientific research. However, conducting research in a controlled laboratory setting requires a large time and money commitment because many studies are experimental. However, using strong AI systems could alleviate these difficulties [38–43]. In data analysis, a wide range of techniques and strategies are used to make forecasts, with ongoing improvements and additions to these tools being common. Within the topic of machine learning, numerous subsets may be made from the techniques provided here. Many different mathematical techniques are covered by these subgroups, including algorithms, linear equations, and polynomials with one or more variables. However, given their inherent complexity, it's crucial to recognize that these techniques could not be as straightforward as supervised algorithms and other neural networks. In this study, in the most basic section, 20 machine learning algorithms (MLA) are introduced. Then 10 indicators of machine learning algorithm evaluation are introduced. The data analysis is then handled initially to ascertain how the input data behaves on the result. The data is then sent to machine learning algorithms and predictive operations are performed. Finally, using evaluation indicators and data analysis charts, the best machine learning algorithm and its mathematical equations are extracted. The schematic of the stages of this study is according to Fig. 1.

This research utilizes machine learning methods for several key reasons. Various parameters, including the SVF (φ), temperature (T), and shear rate (γ), influence the rheological behavior and viscosity of non-Newtonian hybrid nanofluids, such as nanofluids containing iron, copper oxide, water, and ethylene glycol. The viscosity is influenced in a complex, nonlinear manner by these parameters, making it challenging for traditional modeling methods to accurately represent this behavior. The capacity of machine learning algorithms to describe complicated and nonlinear connections between variables makes them ideal instruments for precise viscosity prediction. It is time-consuming and expensive to conduct experimental procedures to assess the viscosity of nanofluids under various circumstances. To save time and money, machine learning algorithms can precisely estimate viscosity, which means fewer tests are needed. Machine learning algorithms, especially more complex ones like artificial neural networks (ANN) and multivariate polynomial regression (MPR), may learn from experimental data and provide incredibly precise predictions. This degree of accuracy is required for engineering equipment design and heat transfer system optimization. Several dimensions and parameters, such as temperature, shear rate, and SVF, are frequently included in nanofluid data. Machine learning algorithms may discover relationships that were previously undiscovered by examining this enormous volume of data. Because machine learning algorithms are flexible, they may be adjusted to accommodate new data or shifting test conditions. Examples of actual industrial applications that considerably benefit from these techniques are the design of heat exchangers and cooling systems. With accurate estimates of nanofluid viscosity, engineers may more effectively design heat transfer systems. As a result, engineering systems operate better overall and use less energy and fuel. By using different evaluation criteria, such as mean square error (MSE), coefficient of determination

(R^2), and mean absolute error (MAE), the performance of multiple algorithms can be compared, and the best approach for viscosity prediction can be selected. This comparison can be used by researchers to choose the most accurate and error-free model for practical uses. The results of this work could be expanded upon by researchers who are interested in heat transfer systems and nanofluids. Additionally, the provided models may be advantageous for both technical equipment and cutting-edge cooling systems in the workplace. The intricacy of nanofluid behavior, the need to reduce the time and cost of experimental tests, the aim to increase prediction accuracy, the ability to handle data with multiple dimensions, the study's adaptability, and the study's real-world industrial applications all make the use of machine learning algorithms in this study completely justified. These techniques not only improve viscosity prediction but also offer a workable solution for engineering equipment design and heat transfer system optimization.

The present work adopts a novel approach by leveraging 20 different machine learning algorithms for the prediction of non-Newtonian hybrid nanofluids viscosity. These algorithms are selected based on their strengths and applications, as they improve prediction accuracy and offer insights into the complex relationships between the input variables (solid volume fraction (SVF), temperature and shear rate) and the target variable (viscosity). The selected algorithms include a variety of machine learning techniques, such as regression models, e.g. Ridge Regression, Lasso Regression, and Elastic Component Regression (ECR), tree-based models, e.g. Decision Tree (DT) and XGBoost (XGB), neural networks (e.g. Multi-Layer Perceptron (MLP) and Back Propagation Neural Network (BPNN)) and ensemble methods (e.g. Gaussian Process Regression (GPR) and Support Vector Machine (SVM)). This makes both linear and nonlinear relationships in the data explored. The rheological behavior of non-Newtonian hybrid nanofluids is complex and nonlinear as it depends on various parameters. The intricate interactions observed in these datasets can be identified by algorithms like Multivariate Polynomial Regression (MPR), which outperformed others in our test as they can model polynomial behavior between inputs and outputs. Algorithms like MPR, ANFIS (Adaptive Neuro-Fuzzy Inference System), and GMDH (Group Method of Data Handling) had prevailed in previous studies with similar datasets, providing accurate and robust predictions with computational efficiency. The study compares algorithms by testing 20 algorithms, providing researchers with a comprehensive benchmark for relatively simple and advanced algorithms and highlighting the most suitable algorithm(s) for any application. We centered the evaluation of the performance of each algorithm on ten metrics including, but not limited to, Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Coefficient of Determination (R^2), and metrics related to recall, precision, f-measure. This thorough evaluation not only highlights the best algorithm but also its strengths, weaknesses, and where improvements can be made. The algorithms are commonly adopted for industrial cases such as heat transfer systems and cooling technologies. SVM and ANN models' expressiveness has made them popular choices for predicting nanofluids' thermal properties, making them relevant for this study. Moreover, MPR and DT provide interpretable results via mathematical equations and decision rules, respectively, which is key for engineers and scientists who want to know which mechanisms affect viscosity. Thus, the choice of these 20 algorithms was motivated by their potential to accurately and consistently predict viscosity for hybrid nanofluids while also providing a benchmark for further research in the area. Overall, the ranges do a thorough job of exploring the published requirements, and emphasizing the balance of the MPR as the best algorithm – this says a lot about why it is vital to choose the correct tool for the job. Making selections will become even better in future work as more influences like computational cost and scalability are added.

2.1. MLA

Ridge Regression cannot be discussed unless regularization is defined. The phenomenon of both underfitting and overfitting is well-known to occur in the multiple regression area. Making regulations is a practical way to handle and lessen these problems. This entails measuring the regression model's parameters and penalizing them to get an optimal value. The result is a reduction in complexity while maintaining the model's efficacy. Under these conditions, the decision is fairly favorable.

- Remarkably high number of descriptive variables
- Because of their enormous quantity, there are more variables than observations.
- The descriptive variables exhibit one or more collinearities.

A "ridge regression" is a sort of linear regression model whereby the coefficients are not estimated using ordinary least squares (OLS) but rather a biassed estimator known as the "ridge estimator," with a reduced variance than the OLS estimator. Sometimes the mean squared error of the ridge estimator—the result of its variance times square of bias—is less than the OLS estimator's. Ridge Regression and Lasso Regression punish the absolute value of the regression coefficients in somewhat similar ways. Another name for Lasso regression is L1 regularization, a frequently used technique in statistical modeling and machine learning that forecasts and assesses the correlations between variables. The basic goal of LASSO regression is to compromise model accuracy with simplicity. This is achieved by adding a penalty component to the traditional linear regression model, therefore encouraging sparse solutions whereby some coefficients must be exactly zero. Lasso is quite useful for feature selection because of this quality since it can automatically identify and remove extraneous or redundant variables. Lasso and Ridge Regression are used in the linear regression method known as ECR. ECR incorporates a regularization factor and, like Lasso and Ridge Regression, seeks to minimize the sum of squared errors between observed and predicted values. But what distinguishes ECR is its combination of L1 and L2 regularization methods. PLR is a covariance-based regression technique that is quick, effective, and ideal. It is advised in regression scenarios with a high number of explanatory factors and a high probability of multicollinearity, or the correlation between the explanatory variables. The PLR is a technique that narrows down the collection of predictors from the original set of variables. A regression is then carried out using these predictors. Similar to k-Nearest Neighbors, GPR makes predictions using training data. It delivers a prediction together with a quantification of uncertainty and performs well with tiny data sets. It is necessary to specify the previous mean and prior covariance. It specifies the covariance as a kernel object. One kind of probabilistic model that can be applied to regression tasks is the GPR model. This non-parametric approach is predicated on the idea that the function that needs to be learned comes from a Gaussian process. The model can produce predictions with a well-defined uncertainty thanks to this assumption, which is helpful for tasks like active learning and uncertainty-aware decision-making. Multiple regression, or MLR for short, is a statistical method that predicts a response variable's value using several explanatory variables. Modeling the linear relationship between the explanatory (independent) factors and response (dependent) variables is the aim of multiple linear regression. Since multiple regression uses more than one explanatory variable, it is essentially an extension of ordinary least-squares (OLS) regression. MPR, sometimes referred to as simple polynomial regression, can be used to examine one variable in a univariate scenario. Nevertheless, multiple polynomial regression allows it to be applied to analyses involving many variables. The structure of real trees, which have a root that spreads out until it reaches a leaf, is the model for DT algorithms. The above-described method is widely used for both the classification of discrete and continuous data, as well as for regression methods that use real or continuous input data [44,45]. The reduction of the subgroups is the first requirement for data segmentation. This technique performs well when dealing with complex data; but, when the data grows in size, more

branches and leaves develop in the data, causing the data processing to slow down. A method for reducing the leaf count was added to the algorithm to circumvent this issue. Other methods could be used, like branch trimming and segmenting the total number of branches. In machine learning, support vector machines (SVM) [46,47] are frequently utilized for classification tasks. However, SVM may also be utilized for regression tasks using Support Vector Regression (SVR). SVR and SVM share similar principles, but instead of classifying data points, SVR concentrates on forecasting continuous outputs. In this lesson, the principles of SVR will be examined, with a focus on sigmoid kernels, and quadratic, and radial basis functions. SVR is capable of handling complex, non-linear relationships in data by utilizing these kernels. One of the key varieties of deep neural networks is multi-layer neural networks, or MLPs (Multi-Layer Perceptrons), which include three levels minimum: an input layer, a hidden layer (or hidden layers), and an output layer. Each neuron, or unit, in these networks, is made up of several different weights and is in charge of translating input into output. MLP consists mostly of the following elements.

- Despite its multi-layered architecture and great number of neurons, MLP can learn intricate and non-linear patterns [48].
- MLP can fit fresh data and environmental changes [49].
- MLP moves data across layers using activation functions including sigmoid or ReLU [50].
- The MLP training method uses [51] the error backpropagation method to update network weights. Ultimately, MLP is a useful neural network model capable of spotting complex patterns and applied for a variety of problems including picture recognition and price prediction.

A machine learning method functioning as a nonlinear model is the Radial Basis Function (RBF) algorithm [52,53]. This method rests on radial basis functions, used as activation functions in the buried layers of the neural network. Important features of the RBF algorithm are as follows.

- Radial Basis Functions: RBF uses the Gaussian function among others to translate input to output.
- Capacity to learn intricate patterns: radial basis functions let RBF learn non-linear and complex patterns.
- Dimension reduction technique: RBF usually reduces input dimensions and increases algorithm efficiency by using PCA or another dimension-reducing technique.
- Usually utilized in RBF training are adaptive techniques including the adaptive nearest neighbor (KNN) algorithm.

Simply said, the RBF approach is a non-linear machine learning method based on radial basis functions to address challenging pattern issues. Rapid and efficient training of neural networks is accomplished using an Extreme Learning Machine, or ELM, machine learning method. This method consists of a single hidden layer whose weights are selected at random then support vector amplification (SVR) or linear regression to maximize the weights of the output layer. The main elements of the ELM algorithm are.

- Fast training: ELM can train models quickly due to its use of random weights and a short training procedure.
- High efficiency: By employing the support vector amplification technique or linear regression, ELM can forecast and adapt to new data with high efficiency.
- ELM runs automatically and does not need the configuration of challenging factors like the number of layers or neurons.
- Reiterability in several issues: ELM is simple and efficient so it may be used in many different fields, from prediction to pattern recognition.

The minimum value of an objective function is found using a gradient descent optimizing method. The gradient reduction or inverse gradient reduction principle—which holds that the algorithm moves in the direction of the inverse gradient of the objective function at each step—defines the basis of this method's operation. The salient features of the gradient descent method are as follows.

- Ability and simplicity implementation: Relatively simple on this algorithm to be able to understand and easy to implement.
- Optimization efficiency: Gradient descent is a powerful optimization technique for finding the minimum of highly complex functions.
- Broad Range of Services: This technique is used in the fields of machine learning, neural networks, parameter optimization, and optimization problems.
- Local minimum: One of the problems with this algorithm is that it often gets stuck at local minimums. This is corrected by applying better heuristics like stochastic gradient descent.

The gradient Descent method is a powerful optimization technique and is used extensively for parameter optimization and in machine learning. The least value of an objective function is what we want to find. To solve the parameter estimation optimization problem presented in the model, a nonlinear optimization algorithm, the Levenberg-Marquardt method, is employed. This method builds upon the gradient Lloyd method to overcome issues of fitting parameters to data. The features of the Levenberg-Marquardt algorithm are as follows.

- *Great efficiency: This algorithm is highly accurate and fast in getting the solutions to parameter optimization problems.*
- Levenberg-Marquardt is a solution for estimating parameters in nonlinear models and determining nonlinear problems.
- *Stability: As a result of integrating gradient reduction and other techniques, this algorithm solves hard problems with a high degree of stability.*
- *Flexibility: Levenberg-Marquardt models can be improved with new data over time.*

BFGS (for Broyden–Fletcher–Goldfarb–Shanno) is an optimization method for solving unconstrained optimization problems. This strategy requires Quasi-Newton methods which is an upgrade of Newton's method which solves non-linear optimization problems. The essential components of the BFGS algorithm are as follows.

- *High efficiency: BFGS is among the high-performance algorithms to solve parameter optimization problems with high speed and accuracy.*
- *Application in non-linear problems: This is the second area in which, estimating parameters in non-linear models and solving non-linear optimization problems is performed efficiently.*
- *Stability: BFGS, being a quasi-Newton method, provides a higher level of stability when working with hard-to-solve problems.*
- *Customization: As time passes, the algorithm can optimize its models and make needed modifications to the data sets.*

Thus, the BFGS algorithm is an Unconstrained optimization algorithm that is configured based on the Quasi-Newton method to optimize parameters. Hence, due to its excellent efficiency, stability, and adaptability, it can be employed to address complex problems. A backpropagation perceptron neural network (BPNN) is an artificial neural network that takes in data to solve classification and prediction problems. This method utilizes a multilayer neural network (MLP) architecture which contains at least three layers: an input layer, a hidden layer, and an output layer. The key ingredients of BPNN algorithm are as follows.

- *Capacity for learning: BPNN can learn complex patterns and relationships in data to improve prediction.*

- *Feature Selection: A Large Amount of Flexibility in the tuning of the algorithm can be applied to a given problem (or challenge) & alteration in the data.*
- *Parallelism: The ability of BPNN to process information in parallel allows for rapid training and prediction.*
- *Wide application: This method is used in multiple domains like pattern recognition, image processing, text processing, and data analysis.*

The ability to carry out nonlinear mapping through multilayer networks with learning, adaptability, and parallel processing gives the backpropagation algorithm an advantage in classification and prediction problems. XGBoost, a tree-based machine learning method, is used on prediction and classification problems [54,55]. This method uses gradient boosting architecture and offers an improvement over more traditional boosting techniques such as AdaBoost. The main features of the XGBoost algorithm are.

- *XGBoost is one of the machine learning algorithms with excellent performance that might be applied to handle challenging problems due to its remarkable speed and accuracy.*
- *Resistance to overfitting: This method uses reasonable parameter selection and expectations (regularization) as control systems against overfitting.*
- *XGBoost can adapt to many facts and difficulties and generate better forecasts with a great degree of accuracy.*
- *Two traits this method supports are reproducibility and parallel distribution, which define features.*

In conclusion, the XGBoost algorithm is a well-liked machine learning algorithm due to its high performance, resistance to overfitting, adaptability, and support for repeatability features. It is an incremental gradient architecture-based high-performance algorithm for prediction and classification problems. A probabilistic model for modeling and displaying probabilistic interactions between variables is the Bayesian network. Conditional probabilities between variables and prior probabilities are represented in this model by the use of Bayesian probability theory [56–58]. The following are the Bayesian network's key characteristics.

- *Visual representation:* A directed graph with nodes for variables and edges for probabilistic correlations between variables is used to illustrate a Bayesian network.
- Complex probabilistic relationships between variables can be properly displayed and modeled by this approach, which can also fully represent conditional probabilities.
- *Application of Bayesian rules:* To update posterior probabilities in response to fresh data, a Bayesian network employs Bayesian rules.
- *Use in decision-making:* This model works effectively when making decisions in complex and unpredictable situations.

To put it briefly, a Bayesian network is a probabilistic model that represents and models probabilistic relationships between variables using Bayesian probability theory. This model is appropriate for modeling and analyzing data in complicated and uncertain situations because it makes use of visual representation, probabilistic connection modeling, Bayesian rule application, and application in decision-making. The Adaptive Neuro-Fuzzy Inference System, or ANFIS [59,60], is a hybrid model that models and predicts data by combining fuzzy systems and neural networks [61]. This algorithm can learn from data and is automatically adaptive [62]. The following are ANFIS's key characteristics.

- ANFIS combines fuzzy systems and neural networks: Fuzzy systems allow for the accurate representation of fuzzy concepts, while neural networks can learn complex patterns.
- Flexibility: This approach can self-adjust its parameters in reaction to both data changes and environmental changes.
- It makes accurate predictions: The power of neural networks and fuzzy systems, gives ANFIS the ability to make accurate predictions, regarding data.
- Versatility: The algorithm can be applied in various fields, including control, decision-making, prediction, and optimization.

Simply put, ANFIS is a predictive model that predicts an intermediary through a sum-out of the output fuzzy set from the input neural networks by combining fuzzy systems and Artificial Neural Networks (ANNs). This algorithm finds applications in several areas like as optimization, control and prediction, etc. It is highly adaptive and precise in predicting data. The Group Method of Data Handling, or GMDH algorithm is a machine learning algorithm for data modeling and prediction. This program is used to create complex and lethal prediction models using hybrid techniques. The main components of the GMDH algorithm are.

- Creating groups helps GMDH to create complex and all-encompassing models. This approach creates prediction models by aggregating many feature groups gradually.
  GMDH can automatically update its settings and adjust to changes in data and other situations.
- Accurate prediction: This algorithm can forecast data with accuracy utilizing group building and algorithms combined.
- Wide application for GMDH spans control, optimization, prediction, and decision-making among other fields.

The GMDH algorithm is, all things considered, a machine-learning method that groups data to create sophisticated and exact prediction models. Wide-ranging applications for this method abound in many different industries; it also has the ability to learn dynamically and yield precise forecasts.

2.2. Tr and HP

Setting some parameters (like the activation function) to designated values helps a machine learning model to generalize to different data usage patterns [48–50,63]. The goal is to optimize the model's solution by adjusting these parameters, also known as hyperparameters. The selection pressure coefficient, neuron count, and number of layers in a GMDH neural network are referred to as hyper-parameters. The optimal number of neurons and layers, as well as the activation function of the hidden layers for the LM, BFGS, BPNN, MLP, BN, and GD algorithms, are *tansig* and *purelin*, respectively. The kind and quantity of membership functions affect the ANFIS algorithm's performance. The SVM algorithm is a polynomial kernel function type, and its HP support vectors are determined by the epsilon value. HPs for other algorithms are set in the same manner. Seventy percent of the data are utilized as Tr data to begin the regression, with the remaining portion being used as test and validation data. Aiming to estimate the objective functions, different regressions use different paths and establish various mathematical relationships between the input variables to do so.

2.3. EC

Regression algorithms' performance needs to be assessed using a range of indicators. The NSE, MARE, R, MSE, R^2, MAPE, MAE, RMSE, MBE, and CvRMSE were the ten criteria that were used in this study. The MSE quantifies how far the model deviates from actual values. This criterion aids in selecting the ideal model. A lower mean square error (MSE) indicates that the model is doing better when there is less of a difference between the predicted and actual outputs. The RMSE criteria is the square root of the error that is obtained by taking the square root of the MSE. Although MAE and MSE are similar, MAE calculates the error's absolute value as opposed to its mean squared error. MAE offers a more accurate picture of the overall error since it rates mistakes

differently than MSE. Dividing the mistake by the real amount and then multiplying the outcome by 100 yields the MAPE—a percentage variance. This percentage technique offers a consistent image of the expected mistakes and helps one understand more easily. MAPE gives a more exact estimate of the error rate than MSE. The linear correlation between two variables is measured by R^2. It computes the fraction of the dependent variable's fluctuations that might be ascribed to the independent variable. In currently in-use definitions, R^2 is referred to as the coefficient of determination or the coefficient of detection. Thus, a high or low correlation coefficient indicates a strong or weak relationship between the input variables and the goal functions. The lack of a link is indicated by a number close to 0. The Nash-Sutcliffe efficiency is calculated as the ratio of the observed time-series variance divided by the variance of the modeled time-series error variance. The Nash-Sutcliffe Efficiency that arises when a model is perfect and has no estimated error variance is 1 (NSE = 1). An indicator of the average bias in a model's predictions is the MBE. Even while MBE isn't usually used as the only way to assess model error since it might not be able to catch very high individual prediction errors, it's important for determining and measuring the average bias in the model's outputs. A variable with positive bias represents an overestimation of data from datasets, whereas a variable with negative bias represents an underestimation. Combining the evaluation of MBE with other metrics, such as correlation coefficients, contributes to a more thorough comprehension of model performance. Better model accuracy is shown by lower error values and greater correlation coefficients, especially for directional variables. The coefficient of the Variation of the Root Mean Square Error is referred to as CVRMSE. When calibrating models for measured nanofluid performance, the CVRMSE is employed. This statistic reveals the unstable observed correlation between the baseline period's variables. It is the coefficient of variance of the expected input series concerning the observed one. If the CV(RMSE) is 10 %, the average distance between a point and the fit line is 10 % of the fit line. The Efficiency Valuation Organization (EVO) advises that for linear regressions the coefficient of determination (RMSE) should be less than half of the expected savings fraction. Equations 29–38 [64,65] form the mathematical formulations for these evaluation indices.

$$R = \frac{\sum_{i=1}^{n}\left(y_{i,Experiment} - \bar{y}_{i,Experiment}\right)\left(y_{i,predicte} - \bar{y}_{i,predicte}\right)}{\sqrt{\sum_{i=1}^{n}\left(y_{i,Experiment} - \bar{y}_{i,Experiment}\right)^2 \sum_{i=1}^{n}\left(y_{i,predicte} - \bar{y}_{i,predicte}\right)^2}} \tag{29}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}\left(y_{i,predicte} - y_{i,Experiment}\right)^2}{n}} \tag{30}$$

$$MAE = \frac{\sum_{i=1}^{n}\left|y_{i,predicte} - y_{i,Experiment}\right|}{n} \tag{31}$$

$$MSE = \frac{\sum_{i=1}^{n}\left(y_{i,predicte} - y_{i,Experiment}\right)^2}{n} \tag{32}$$

$$R^2 = 1 - \sum_{i=1}^{n}\frac{\left(y_{i,predicte} - y_{i,Experiment}\right)^2}{{y_{i,Experiment}}^2} \tag{33}$$

$$MAPE = \frac{\sum_{i=1}^{n}\left|\frac{y_{i,predicte} - y_{i,Experiment}}{y_{i,Experiment}}\right|}{n} * 100 \tag{34}$$

$$NSE = 1 - \frac{\sum_{i=1}^{n}\left(y_{i,Experiment} - y_{i,predicte}\right)^2}{\sum_{i=1}^{n}\left(y_{i,Experiment} - \bar{y}_{i,Experiment}\right)^2} \tag{35}$$

$$MARE = \frac{1}{n} * \sum_{i=1}^{n}\left|\frac{y_{i,pred} - y_{i,Experiment}}{y_{i,Experiment}}\right| \tag{36}$$

$$CvRMSE = \frac{RMSE}{\bar{y}_{i,Experiment}} * 100 \tag{37}$$

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

2.4. AP

Taylor diagram is one of the most significant and generally implemented tools in scientific and engineering data analysis, thus making an important contribution in forecasting the behavior of nanofluids. Such a diagram is often utilized by researchers when investigating properties of nanofluids that have gained special interest on account of their distinct thermal and physical properties in a wide array of applications ranging from electronics to medicine and the energy sectors. There are multiple reasons why the Taylor diagram is highly significant for the prediction of nanofluids.

- **Very accurate and efficient:** Taylor's diagram provides a compound representation of multiple statistics such as the correlation coefficient, standard deviation, and RMSD (root mean square error) with high accuracy in the evaluation of forecasting models. This feature allows researchers to further assess their model performance more thoroughly and accurately.
- **Integrated visualization:** A key characteristic of Taylor's diagram is the ability of integrated visualization. This graph allows you not only to compare both models visually but also to observe differences and similarities. This ability is particularly advantageous in studies of nanofluids, which contain many complexities.
- Ability to compare models: Taylor's chart offers a comparison of different forecasting models. By using this tool, researchers can pinpoint and fine-tune the models that yield the most accurate predictions. This is a great success in improving the accuracy of predictions for the type of nanofluid behavior.
- **Other applications:** Besides using the Taylor diagram for predictions about various nanofluid thermal and physical behavior, applications in the simulation of fluid flows, heat transfer analysis and other complex systems also utilize this method. The spectrum of these applications demonstrates the significance and high potential of this tool in the research for nanofluids.

The stability of the Taylor diagram [66] requires the geometric relationship of RMSD, STD, and R. The RMSD indices and standard deviation are retrieved using Equations (39) and (40) [66].

$${\sigma_{ANN,Exp}}^2 = \frac{1}{N}\sum_{n=1}^{N}\left({y_{pred,Exp}}^{(i)} - \overline{{y_{pred,Exp}}^{(i)}}\right)^2 \tag{39}$$

$$RMSD^2 = {\sigma_{Exp}}^2 + {\sigma_{pred}}^2 - 2\ \sigma_{Exp}\sigma_{pred}R \tag{40}$$

The Taylor diagram comes in two varieties [66]: a semicircle that displays both positive and negative correlation, and a quarter circle that only shows positive correlation. In both cases, the R values are represented by the circle's radius on its arc, and the STD values are represented by concentric circles in the circle's center. The RMSD values are displayed on the faces of concentric circles drawn concerning the horizontal reference point on the axis of the hollow circle. The reference point is the value of the experimental data as shown by the STD. The image indicates that the position of the data is determined by the RMSD, the STD, and the R between the analyzed and experimental data. Any experimental data that is found nearer the diagram's reference point is considered to be more precise when determining its value.

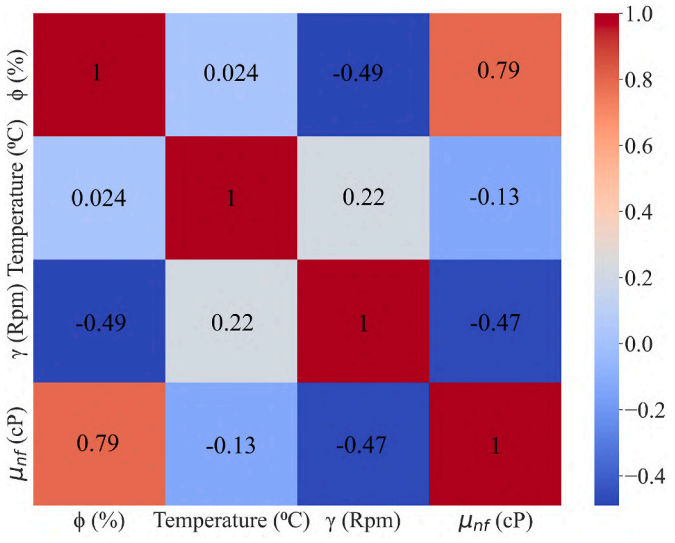


Fig. 2. Diagramming data input and output behavior with a heatmap.

The following formula can be used to compute the μ_{nf} error analysis to assess the precision of values predicted by MLA [67]:

$$Error = \mu_{nf_{Exp}} - \mu_{nf_{pred}} \tag{41}$$

Another graph is the error histogram. The higher the frequency of the error column is closer to zero, which means that the machine learning algorithm has been able to perform the prediction well.

3. Results and discussion

3.1. The study of data behavior

This study incorporates the method and findings of Bahrami et al. [68]. Fe–CuO/water-ethylene glycol hybrid NFs flow was all thoroughly investigated in their thermophysical property studies. Important metrics such as DV is tested within the 25–50 °C temperature range, shear rate (γ) range of 3–100 Rpm and VF range of 0–1.5 %. To be more specific, 204 points were taken from Bahrami et al.'s experimental dataset [68].

This section of the work explores the behavior of the dataset used to predict the viscosity of a hybrid nanofluid devoid of Newtonian character. The output variable in the dataset is viscosity (μ_{nf}), which additionally incorporates shear rate (γ), temperature (T), and SVF(φ). Analysis of this data is necessary for numerous reasons.

- The study particularly shows how the input variables are linked with the output variable (viscosity) via a heatmap. This helps one to identify the most influencing input parameters on viscosity.
- Box charts are a helpful tool for a better knowledge of how the input elements influence viscosity. These graphs show the effects on the viscosity of the nanofluid of temperature, shear rate, and nanoparticle concentration.
- Knowledge of the distributions and correlations of the input variables helps one identify the optimal features to teach machine learning models. In this sense, the models will be more accurate predictors since they might be trained on data relevant for the output.
- Understanding the effect of input parameters on viscosity will help one to maximize the efficiency of nanofluids in useful environments like heat exchangers and cooling systems. Engineers must find the primary factors influencing viscosity if they are to maximize the enhanced thermal properties of nanofluids.

Machine learning techniques must first examine the behavior of the data to forecast the nanofluid set-back. The dataset for this study comprises of the output group itself as well as the input elements influencing behavior of the output. The study outputs μ_{nf} and has φ, T, and γ as its inputs specifically. First of all, one must look at the link between the inputs and outputs. One good way to show the scope of these interactions is with a heat map graph. The output is least affected if a desired input has little to no link with the output. On the other hand, a negative correlation value—that which is near to one—indicates that the input clearly affects the output. Fig. 2 displays the heatmap that illustrates how inputs influence outputs. The correlation matrix between the input variables— SVF (φ), temperature (T), and shear rate (γ)—and the output variable, viscosity (μ_{nf}), is graphically shown in Fig. 2's heatmap. Each pair of variables in the heatmap is given a numerical value and a color gradient, with −1 representing a strong negative connection and +1 representing a strong positive correlation. If the value is positive, then there is a positive correlation between the two variables, and if the value is negative, then there is an inverse correlation, where a reduction in one variable is correlated with a rise in the other. The correlation value of 0.79 indicates that φ has the greatest positive relationship with μ_{nf}, as seen by the heatmap. This proves that the viscosity of the fluid grows exponentially with the SVF. We anticipate this result because the introduction of nanoparticles increases the barrier to flow by strengthening intermolecular forces and obstructing the velocity of the fluid. On the other hand, there is a modest negative association shown by the correlation coefficient of −0.13 between temperature (T) and μ_{nf}. Consistent with the idea that greater temperatures lessen intermolecular tensions and enhance fluidity, the negative sign indicates that viscosity reduces somewhat with increasing temperature. Likewise, γ is negatively correlated with μ_{nf}, with a value of −0.47, indicating that a considerable decrease in viscosity occurs when the shear rate is increased. Because the nanofluid's particles align under shear, the flow resistance is reduced, and this behavior is a reflection of that characteristic. Since temperature and shear rate are inversely related to viscosity, the negative sign in their correlations is a result of this inverse connection. A decrease in viscosity and a weakening of intermolecular cohesion result from an increase in the thermal energy of fluid molecules as the temperature rises. The negative association between shear rate and viscosity is a result of nanoparticle structural rearrangement caused by strong shear, which reduces internal friction.

If the correlation coefficient (R) is between −1 and +1, then the value represents both the strength and direction of the degree of the linear relationship between two random variables. Positive value ($R > 0$) means that as the first variable rises, the second variable rises too, while negative value ($R < 0$) means that as the first variable rises the second variable falls. These characteristics are seen in hybrid nanofluids. The negative correlation between temperature (T) and viscosity ($R = -0.13$) represents the established inverse relationship, wherein the viscosity is known to decline as temperature increases. However, as the temperature increases, due to the increase in thermal energy of the fluid molecules, the attractive forces between the molecules are overcome, thus reducing resistance to flow. The shear rate (γ) is also more strongly inversely correlated with viscosity ($R = -0.47$), which corresponds to the shear-thinning behavior of non-Newtonian fluids. Nanoparticles orient with velocity field and this orientation reduces the internal friction, thus decreasing the viscosity at high shear rates. On the other hand, the volume fraction of nanoparticles, ϕ highly influences the viscosity ($R = 0.79$); as the number of nanoparticles increases, intermolecular interaction and hindrance within the flow path also increase and lead to increasing viscosity.

The heat map of relationships between inputs and outputs is illustrated in Fig. 2, where correlation coefficients between input variables, solid volume fraction (ϕ), temperature (T), and shear rate (γ), and output variable, viscosity (μ_{nf}), are shown. The main outcomes indicate that ϕ correlates positively ($R = 0.79$) with viscosity due to the nanoparticle concentration, which if increased, viscosity substantially. Temperature (T) has a weak inverse correlation ($R = -0.13$), meaning

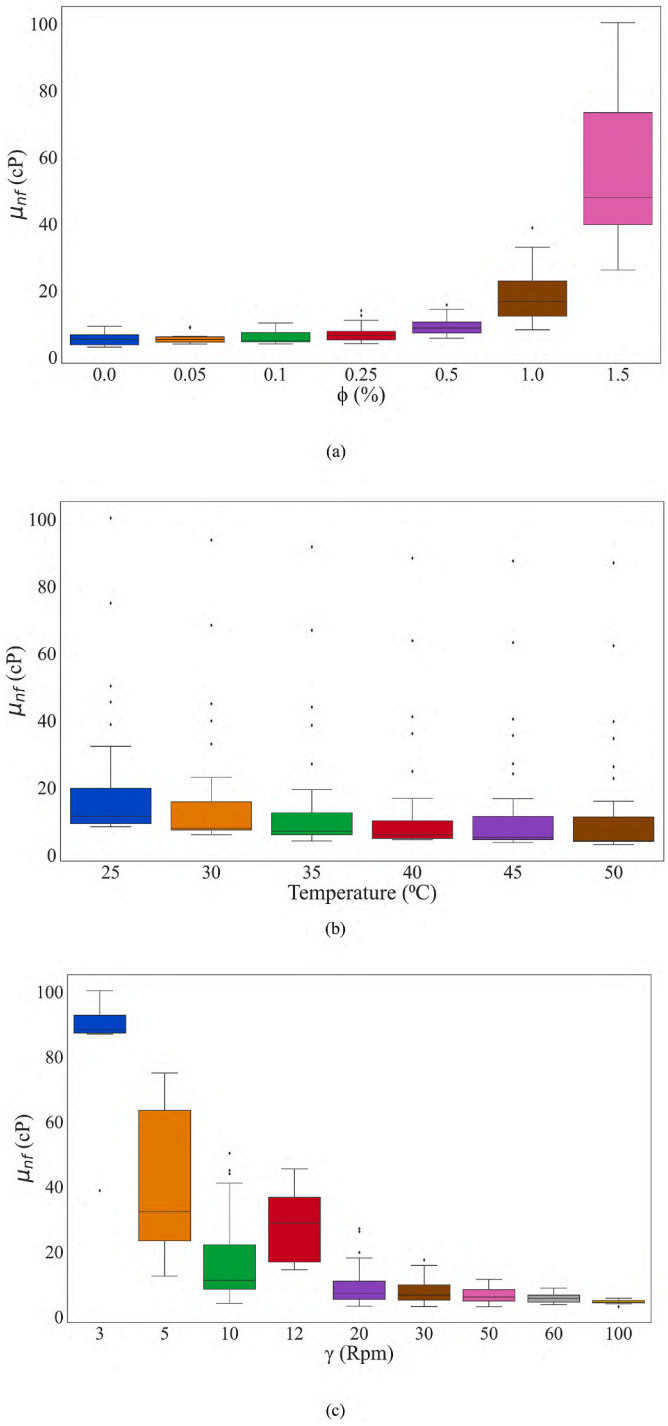


Fig. 3. Box plot to see the behavior of a) φ b) Temperature c) γ

that an increased temperature reduces viscosity by loosening intermolecular forces. Also, shear rate (γ) was negatively correlated (R = −0.47) which is more negative than the second and third ones, but for non-Newton fluids such as paint till now we handle them as Newton fluids to do so when viscosity becomes smaller with the increase of γ. Some ML models work better because they are based on the fundamental properties of the data. For instance, a mad scientist's right algorithm like multivariate polynomial regression (MPR) works well due to its ability to model polynomial relationships between input variables to output variables. In comparison, many non-linear models like elastic component regression (ECR) fail when the interactions are higher-degree, the relationships being considered. To some extent, the non-Newtonian behavior of composite nanofluids is closely related to particle aggregation, Brownian motion, and van der Waals forces, which is an inherent characteristic of composite nanofluids. The MPR is a good model for such data because it allows us to model the complex interactions between these variables. First, the model is grounded in mathematical equations and can be well generalized to the dataset. Simples' linear models would not be able to capture many complicated dependencies between the variables.

According to the sensitivity analysis carried out in this work, shear volume fraction (SVF) has the most impact on viscosity, followed closely by shear rate and then temperature. The heat map (Fig. 2) confirms this result with correlation coefficients. The correlation coefficient between viscosity (μ_{nf}) and nanoparticle volume fraction (ϕ) is 0.79 and with increasing nanoparticle volume fraction, intermolecular forces are strengthened, resulting in increasing viscosity. The shear rate (γ) shows a moderate negative coefficient of correlation of −0.47, in which at higher shear rates the nanoparticles were arranged with the flow and the value of internal friction was less, where viscosity is decreased. Temperature (T) has a correlation coefficient of −0.13; that is, it has a weak positive correlation; Viscosity decreases as temperature increases; As intermolecular cohesion weakens, its effect is small compared with the volume fraction of nanoparticles and shear rate. Thus, the volume fraction of nanoparticles is the dominant factor influencing viscosity, followed by shear rate and temperature.

Fig. 3 shows box plots that show how the input variables, heat (T), shear rate (γ), and SVF, behave and how they affect the output variable, viscosity (μ_{nf}). You may learn a lot about the inputs' effects on the viscosity from box plots, which show the distribution, variability, and relationships of the data. Fig. 3 shows that when the temperature increases, μ_{nf} drops. The reason for this tendency is that when the temperature of the fluid molecules rises, their thermal energy also increases, which weakens the intermolecular interactions and reduces the flow resistance. As a result, the viscosity of the fluid decreases. On the contrary, there is a significant increase in nanofluid viscosity as φ becomes larger. The presence of nanoparticles intensifies the interactions between molecules, leading to the formation of bigger clusters as a result of van der Waals forces. These clusters, in turn, increase the viscosity of the fluid and hinder its movement. It has been noted that the impact of γ on μ_{nf} is not linear. According to the box plot, viscosity initially rises as γ increases, but after reaching an ideal value, it steadily decreases. The shear-thinning behavior of non-Newtonian fluids—in which the alignment of nanoparticles at higher shear rates decreases internal resistance to flow—can be used to explain these phenomena. You can see the data's spread and variability clearly in the box plots as well. The higher variation of viscosity values for φ indicates the high effect of particle concentration on the behavior of the fluids. T has a smaller but significant effect, as indicated by its smaller range and constant decrease in viscosity. Thus, the relationship between γ and viscosity is conditional on the shear rate and other mixing factors, which results in a large data dispersion. These trends illustrate relationships between factors-influencing input and viscosity where φ is the most sensitive input when compared to γ and T, which are less correlated with viscosity; in some cases, the box plots also reflect outliers representing abnormal behavior of viscosity caused by high φ, T, or γ values. This proves the fact that the rheological properties of the nanofluid are nonlinear and complex. To optimize the performance of non-Newtonian hybrid nanofluids in real-world applications, it is vital to consider the viscosity of these fluids, and Fig. 3 shows how these elements interact critically.

These figures demonstrate that the lowest μ_{nf} happens at the highest temperature together with the lowest γ and φ. This phenomenon is explained by the following theories.

- The random movement of nanoparticles inside the base fluid, or Brownian motion, is one of the elements influencing viscosity. The

Table 1
Values of 10 EC for 20 MLA algorithms.

Algorithm	R	R^2	RMSE	NSE	MAE	MBE	MSE	MARE	MAPE	CvRMSE
ANFIS	0.9827	0.9657	3.498	0.975	1.338	−2.132	12.233	0.075	11.5	23.14
MPR	0.9961	0.992	1.66	0.994	1.146	$-5*10^{-12}$	2.753	0.129	7.61	9.98
BN	0.9888	0.9777	2.84	0.983	2.25	5.2	8.056	0.228	25.32	18.775
BFGS	0.9927	0.985	2.28	0.989	1.777	−9.53	5.22	0.198	22.582	15.11
GD	0.9888	0.9776	2.822	0.984	2.15	2.766	7.965	0.223	24.93	18.67
LM	0.9888	0.9778	2.81	0.984	2.194	5.015	7.89	0.228	25.13	18.581
BPNN	0.9087	0.826	7.97	0.87	3.97	−49.89	63.58	0.184	21.143	52.744
DT	0.9808	0.962	3.676	0.972	1.707	$-6*10^{-13}$	13.52	0.083	8.4	24.32
ECR	0.229	0.227	16.6	0.435	10.874	$-2*10^{-12}$	275.567	0.443	112.11	109.81
PLR	0.8069	0.651	11.15	0.75	7.072	$9*10^{-11}$	124.4	0.03	63.23	73.777
ELM	0.9641	0.929	17.06	0.402	16.365	−1631	291.28	0.619	221.76	112.9
GMDH	0.9903	0.981	2.65	0.986	1.941	30.22	7.026	0.19	20.55	17.53
GPR	0.984	0.968	3.4	0.976	1.877	0.393	11.575	0.144	12.49	22.5
Lasso	0.7996	0.64	12.43	0.683	6.714	$-3*10^{-12}$	154.5	0.534	51.17	82.22
MLP	0.9877	0.976	3.01	0.981	2.34	−18.9	9.056	0.23	24.31	19.91
MLR	0.8068	0.651	11.153	0.745	7.071	0.0184	124.395	0.031	63.232	73.777
RBF	0.9913	0.983	2.505	0.987	1.691	19.4	6.27	0.021	18.61	16.57
Ridge	0.9708	0.942	10.283	0.783	4.6	444	105.73	0.291	19.55	68
SVM	0.931	0.867	8.93	0.836	2.703	193.43	79.73	0.099	8.18	59.06
XGB	0.9946	0.942	4.55	0.958	2.48	224.4	20.695	0.166	14.25	30.09

constant collisions between the molecules of the base fluid and the nanoparticles are what create this chaotic motion.

- The addition of the nanoparticles causes the nanomaterials to scatter and aid in the formation of bigger, symmetrical nanoclusters through the base fluid's van der Waals contact with the nanoparticles. Viscosity rises because ethylene-glycol molecules are prevented from rubbing against one another by these nanoclusters.
- Nanostructures lose some of their key properties, such as density, due to their small size and relatively low mass, making them primarily dependent on surface and intermolecular interactions. This results in an extraordinarily high surface-to-volume ratio.
- Viscosity rises in a base fluid containing nanomaterials as intermolecular forces rise.
- Viscosity, a feature of liquids that depends on temperature, is caused by cohesive forces between molecules. A rise in temperature causes liquids to become less viscous. At higher temperatures, liquid molecules have more energy and can overcome the forces of molecular cohesiveness. Consequently, molecules that are energetic move more quickly. The intermolecular connections become weaker and the flow resistance reduces as temperature rises. Newtonian nanofluid viscosity consequently falls as temperature rises.
- It makes sense to consider how the viscosity of the nanofluid is impacted by the temperature-dependent Brownian motion of the nanoparticles.
- Vessel viscosity and flow resistance decrease with increasing temperature because of an increase in the intermolecular distance between the nanoparticles and the base fluid.

Boxplots are presented in Fig. 3 to examine how the input variables (ϕ, T, γ) affect viscosity (μ_{nf}). The results, revealed at the same time, are that the increase of viscosity is qualitative with increasing solid volume fraction (ϕ) and that there is little overlap between levels of ϕ. Rising T uniformly produces lowered viscosity, but T (compared to ϕ) has a much smaller range of behavior. For the shear rate (γ), a nonlinear trend was observed that showed the shear rate first increased and then decreased with viscosity with a specific optimal point, which is in line with the characteristic of nozzle shear of non-Newtonian fluids. Machine learning models capable of modeling multilinear relationships, like multivariate polynomial regression (MPR), Gaussian process regression (GPR), radial basis functions (RBF), for predictions, outperform models constrained to linear assumptions – partial least squares regression (PLR). DTs also perform well because they can split data along thresholds ideal for the piecewise trends observed. Non-Newtonians are known for their complexity; At low shear rates particles may induce clusters that give rise to an increase in viscosity, whereas at high shear rates, alignment of the particles leads to a reduction of drag. Such dynamics require flexible predictors for accurate prediction. Data-wise, ELM (Extreme Learning Machine) and ECR (Elastic Component Regression) are over-simplified (parameter-inefficient), resulting in poor generalizability, while MPR falls somewhere in the middle, able to reach high accuracy whilst also remaining computationally efficient by employing a polynomial model that directly incorporates the governing physical relationships.

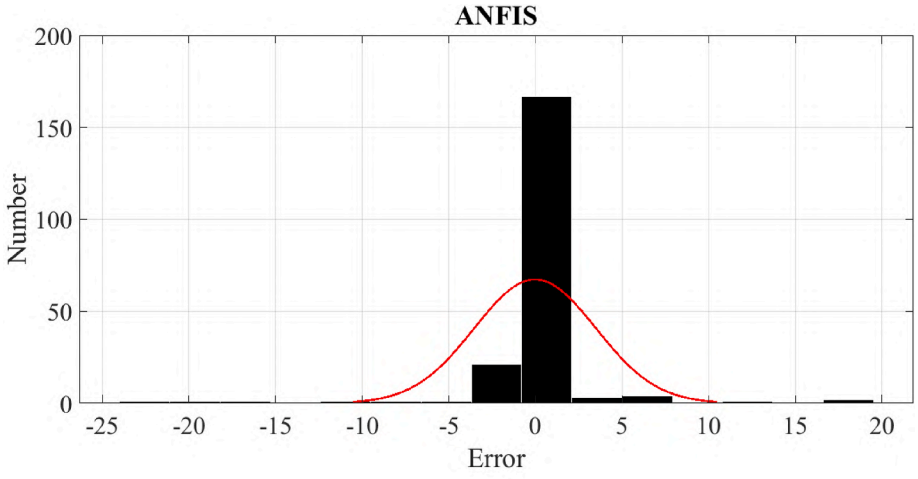
3.2. Results for EC

As mentioned, this study aims to predict laboratory data by machine learning algorithms. For this purpose, 20 machine learning algorithms ANFIS, BFGS, BN, GD, LM, BPNN, DT, ECR, PLR, ELM, GMDH, GPR, Lasso, MLP, MLR, MPR, RBF, Ridge, SVM, and XGB were used to achieve this goal. A 204*4 dataset is introduced to these algorithms, the first 3 columns of which are inputs (T, γ, and φ) and the final column is the target or output function (μ_{nf}). Then the data was divided and 70 % of it was assigned to the training data, 15 % to the validation data and the rest to the test data. In the following, hyperparameters were set and changed for each algorithm to determine the best state of each algorithm. Finally, a machine learning algorithm predicts a value for each data. To evaluate the performance of each algorithm, evaluation indices or EC should be used. In this study, 10 evaluation indicators were used. Table 1 shows the value of these indicators.

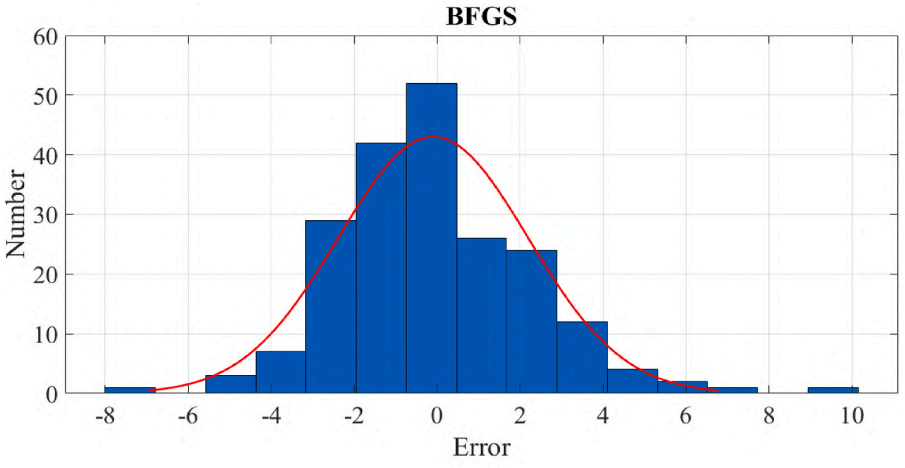
Twenty different machine learning algorithms were tested for their ability to predict the viscosity of non-Newtonian hybrid nanofluids. The results of these tests are shown in Table 1. Several metrics are used to evaluate the models' accuracy and reliability. These include R-squared, R^2, RMSE, NSE, MAE, MSE, MPE, and CvRMSE, among others. With higher numbers indicating greater correlations, the R shows the degree and direction of the linear link between expected and actual values. ECR has the lowest R-value of 0.229, indicating weak correlation, in contrast to MPR, which obtains the greatest R-value of 0.9961, indicating greater prediction accuracy among the methods. R^2 is a measure of how much of the observed data variation can be explained by the model. ECR has the lowest R^2 of 0.227, showing that it cannot adequately predict the viscosity data, while MPR scores best with an R^2 of 0.992, showing that it captures virtually all data variability. Measuring the average size of mistakes, smaller numbers indicate greater performance according to the root mean squared error (RMSE) and the mean squared error (MSE). There is very little discrepancy between the predicted and actual values when using MPR, as its RMSE is 1.66 and its MSE is 2.753. Alternatively,

ECR demonstrates substantial prediction errors with an RMSE of 16.6 and an MSE of 275.567, the highest values. The NSE measures how well the models can predict outcomes in comparison to the average of the data points. With an NSE of 0.994, MPR is a very effective model, but ECR's 0.435 shows that it is not very good at making predictions. As with RMSE, a smaller number indicates better accuracy when calculating the MAE, which gives an average of the absolute errors. The MAE of 1.146 for MPR shows that it makes accurate predictions, but the MAE of 10.874 for ECR shows that it is quite inaccurate. If the MBE is negative, then the predictions are underestimated, and if it is positive, then the predictions are overestimated. This statistic displays the average bias in predictions. While ECR shows a greater bias, MPR provides unbiased predictions with a minimal MBE of $-5*10^{-12}$. One way to compare the performance of different models is using the mean absolute percentage error (MAPE), which represents mistakes as a percentage. In comparison to ECR's 112.11 % MAPE, which indicates exceptionally high prediction errors, MPR's MAPE is the lowest at 7.61 %. As a last metric, the coefficient of variation of RMSE (CvRMSE) indicates the consistency of performance compared to the error; lower values show more stable predictions. With a CvRMSE of 9.98 %, MPR has the greatest prediction accuracy, while ECR's CvRMSE of 109.81 % shows that its forecasts are quite unpredictable. These numbers show that MPR is the top algorithm, providing very reliable and accurate predictions. In comparison to MPR, other algorithms like BFGS, BN, and GPR do decent work; but they do not achieve MPR's level of accuracy. On the other hand, algorithms like ECR, PLR, and ELM perform poorly because they have a hard time accurately modeling the dataset's nonlinear connections. Choosing the right algorithm for predictive modeling is crucial, as shown by MPR's ability to capture complicated dynamics and provide very precise viscosity forecasts.

According to Table 1, the value of R is different for each algorithm and ranges from 0.229 to 0.9961. If this value is close to 1, the best MPR algorithm and the worst one should be related to the ECR algorithm. The value of R^2 is an index like R, whose value varies from 0.227 to 0.992 for algorithms, the best algorithm is MPR and the worst one is ECR algorithm. Also, the NSE index is following these two indices. Other indicators are related to the amount of error between the predicted points and the experimental data, the smaller the value, the better. Only the MPR algorithm has a CvRMSE value below 10 %, which is optimal, and the rest of the algorithms are more than this value, and this index reaches 112.89 for the ELM algorithm, which is the worst value. MSE and RMSE are close indicators and the only difference between them is that one of them is squared. Obviously, if the value of this index is close to zero, the machine learning algorithm has less error, the lowest error is

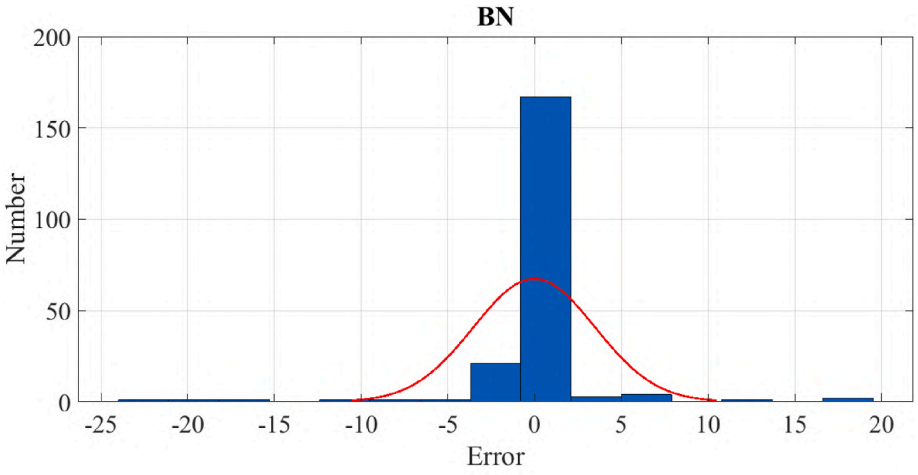


(a)

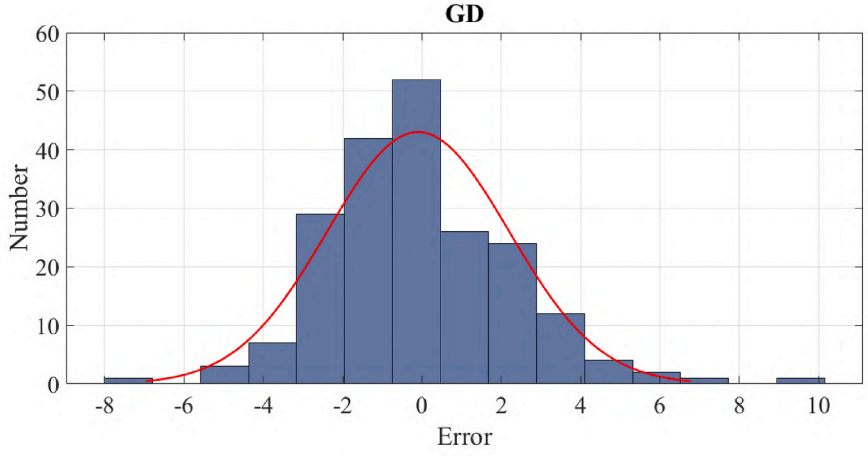


(b)

Fig. 4. Histogram plot error for a) ANFIS b) BFGS c) BN d) GD e) LM f) BPNN g) DT h) ECR i) PLR j) ELM k) GMDH l) GPR m) Lasso n) MLP o) MLR p) MPR q) RBF r) Ridge s) SVM t) XGB algorithm.

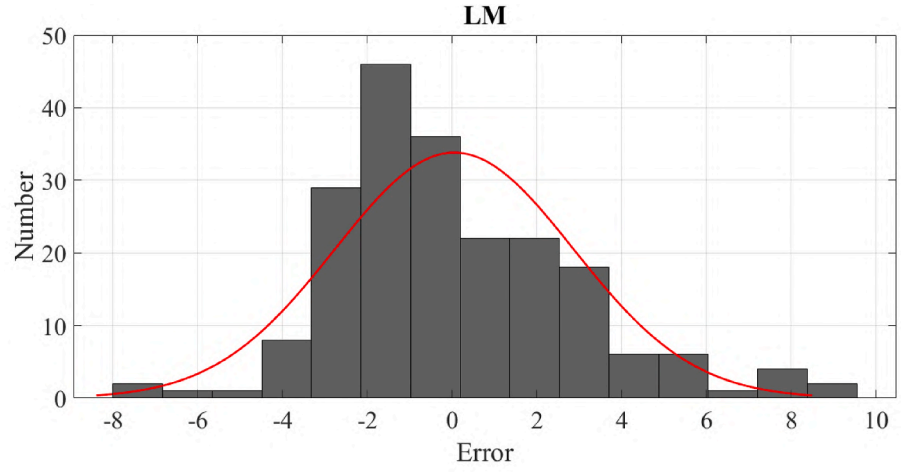


(c)

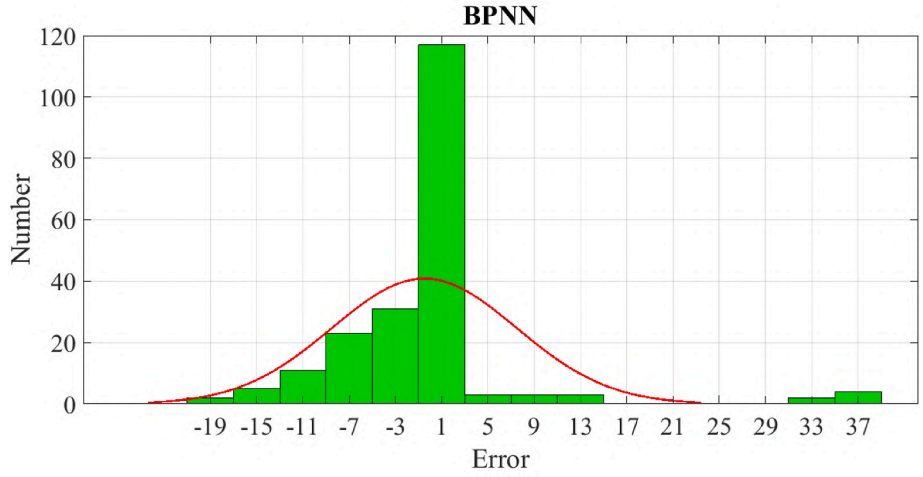


(d)

Fig. 4. (*continued*).

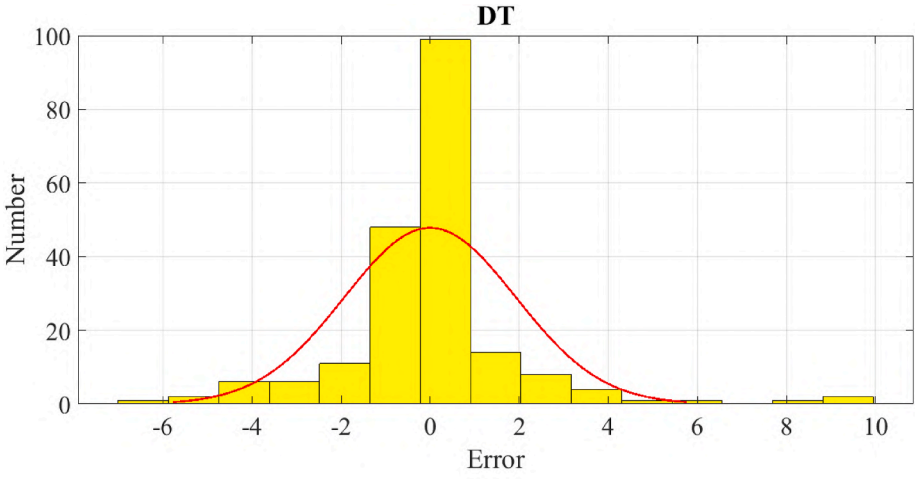


(e)

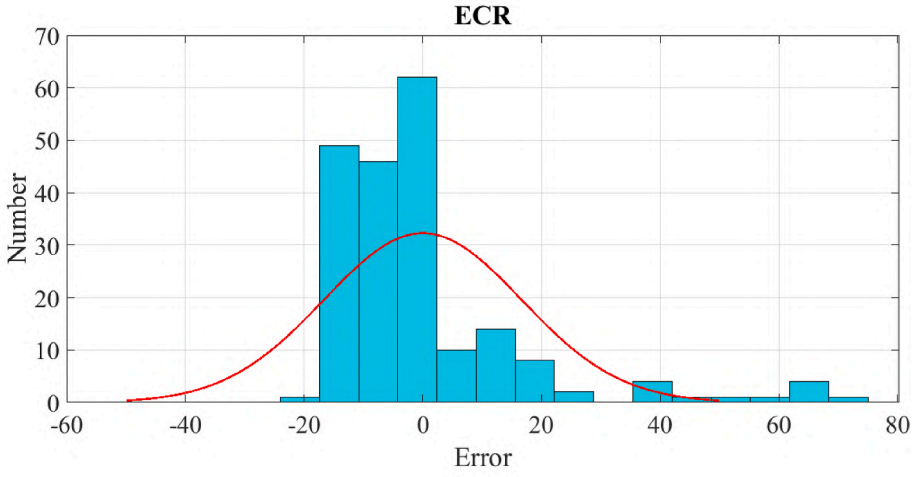


(f)

Fig. 4. (*continued*).

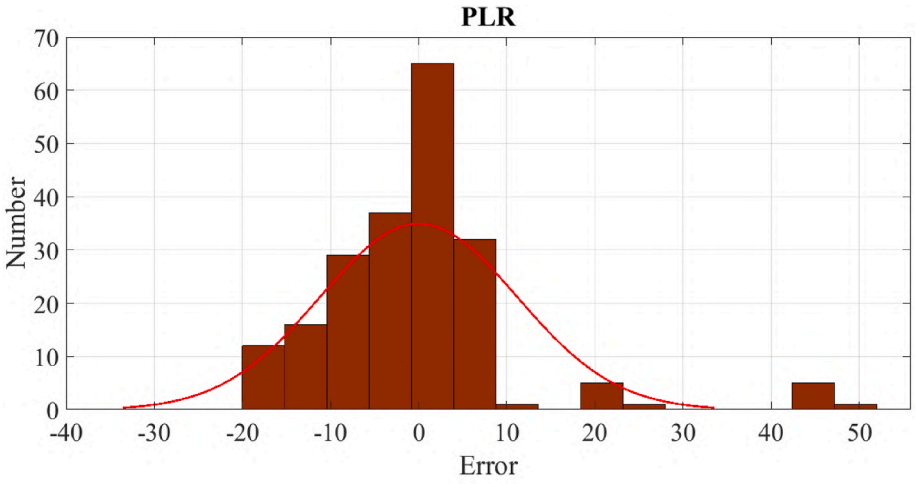


(g)

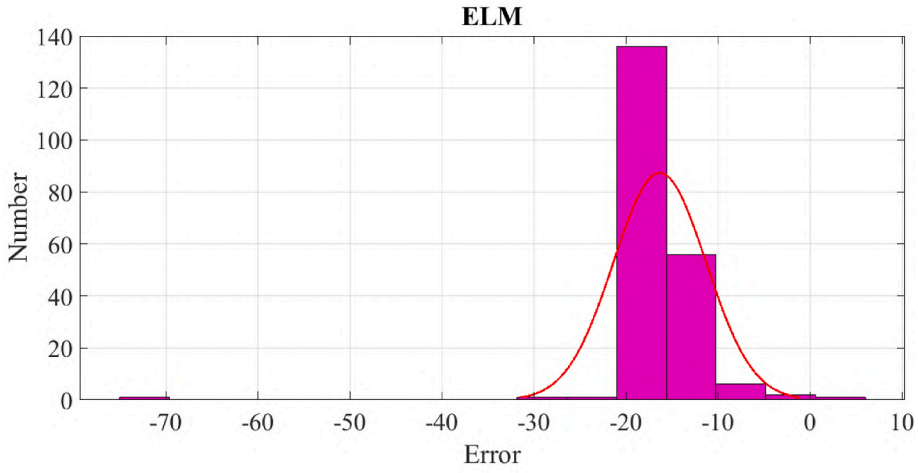


(h)

Fig. 4. (*continued*).

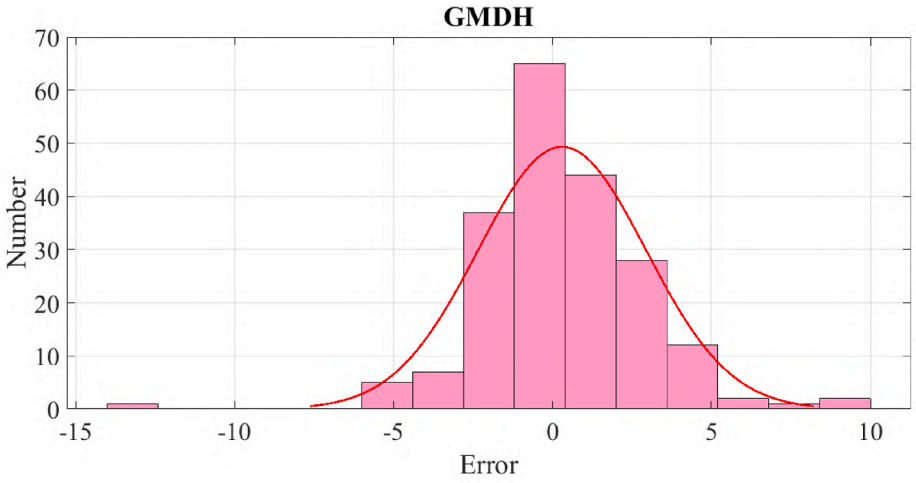


(i)

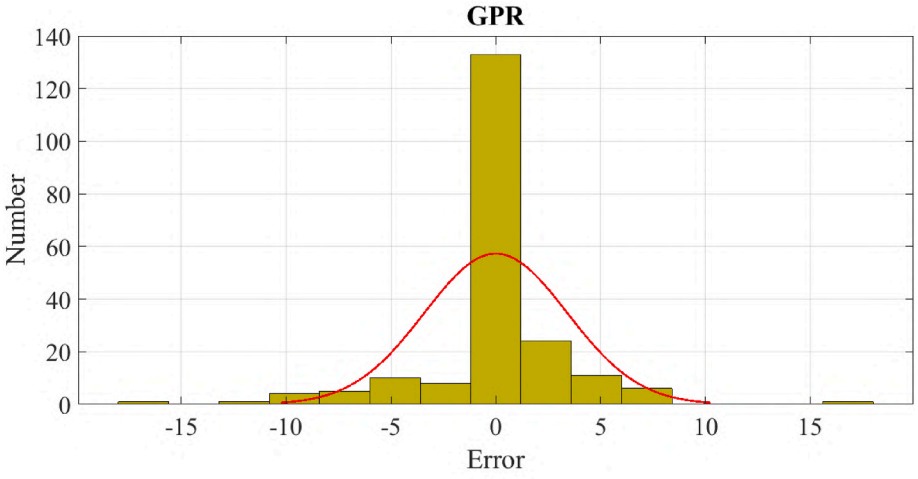


(j)

Fig. 4. (*continued*).

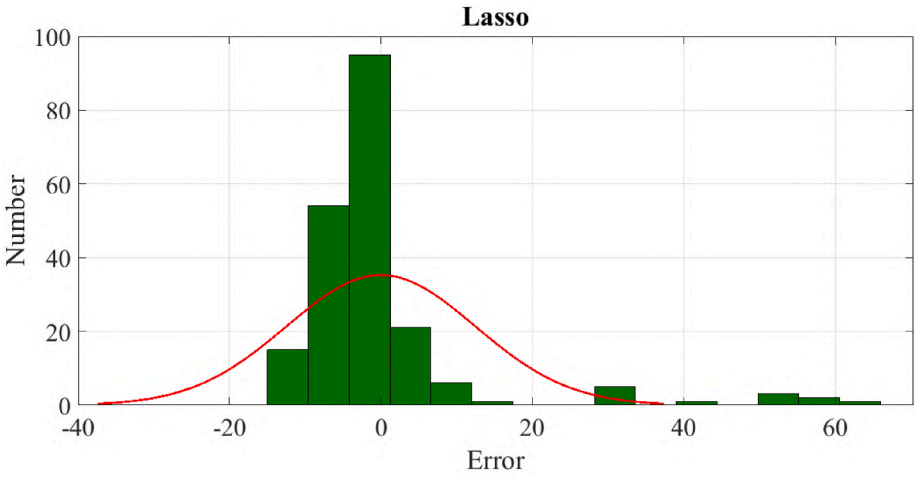


(k)

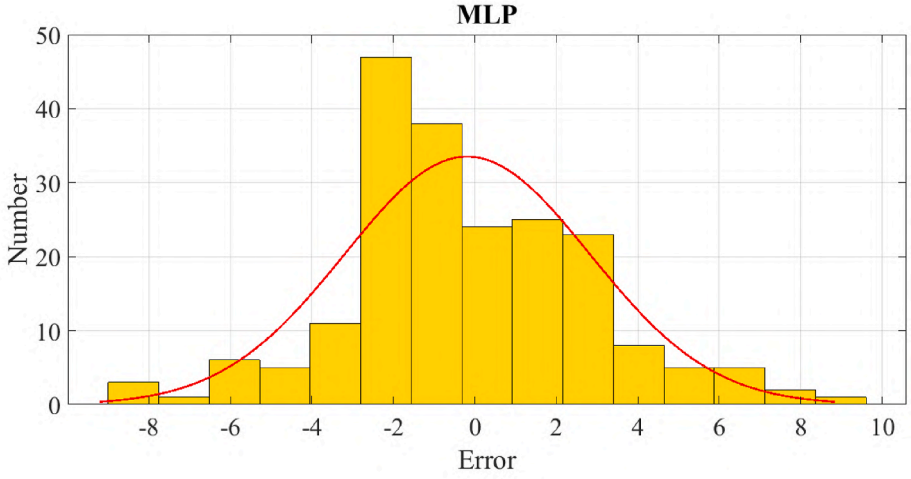


(l)

Fig. 4. (*continued*).

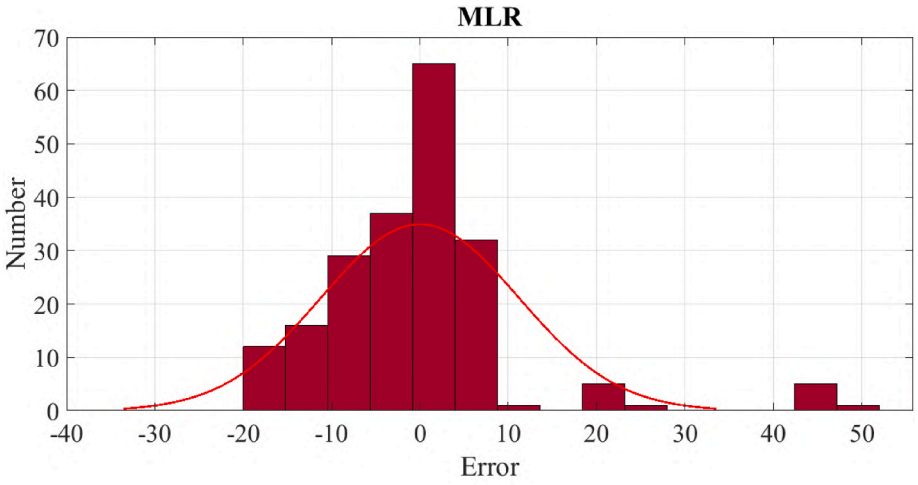


(m)

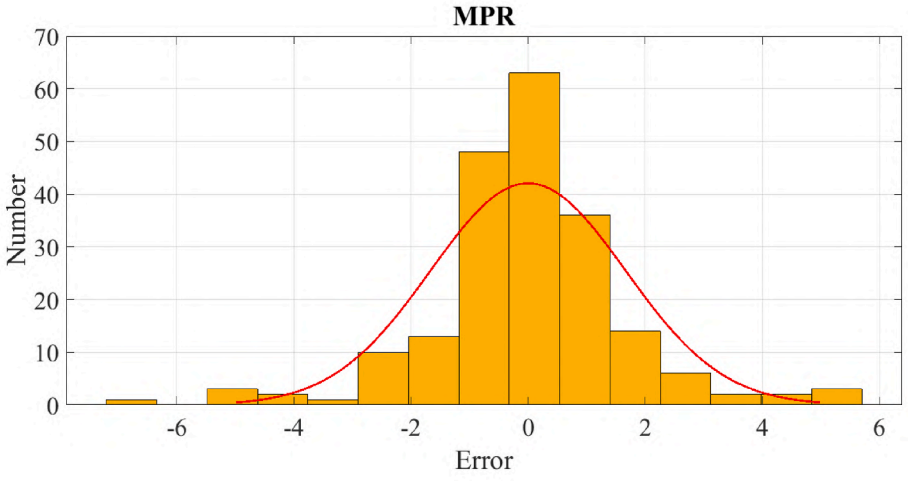


(n)

Fig. 4. (*continued*).

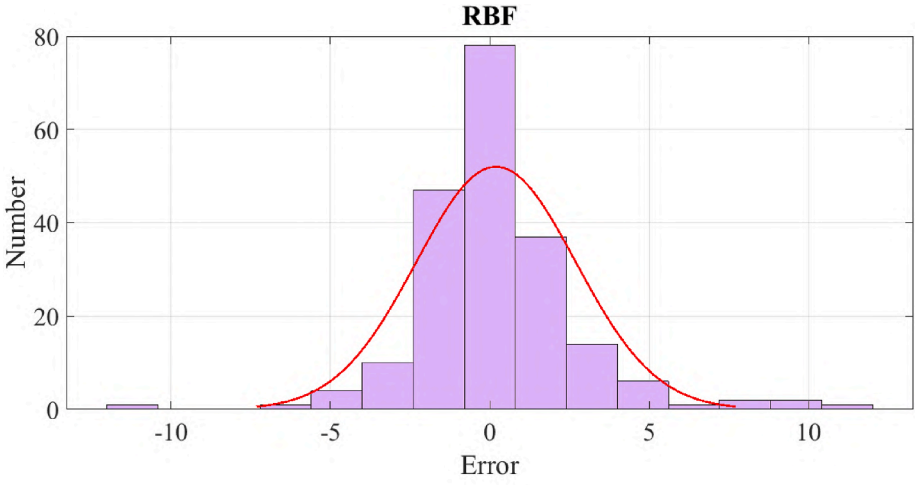


(o)

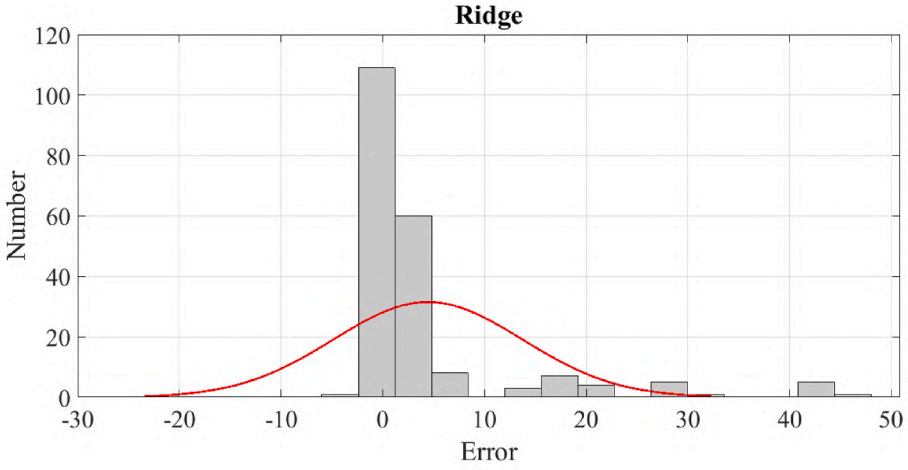


(p)

Fig. 4. (*continued*).

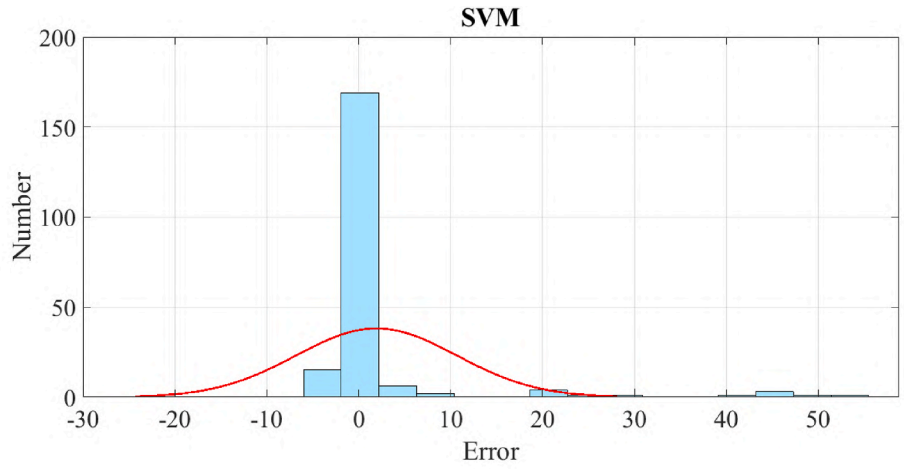


(q)

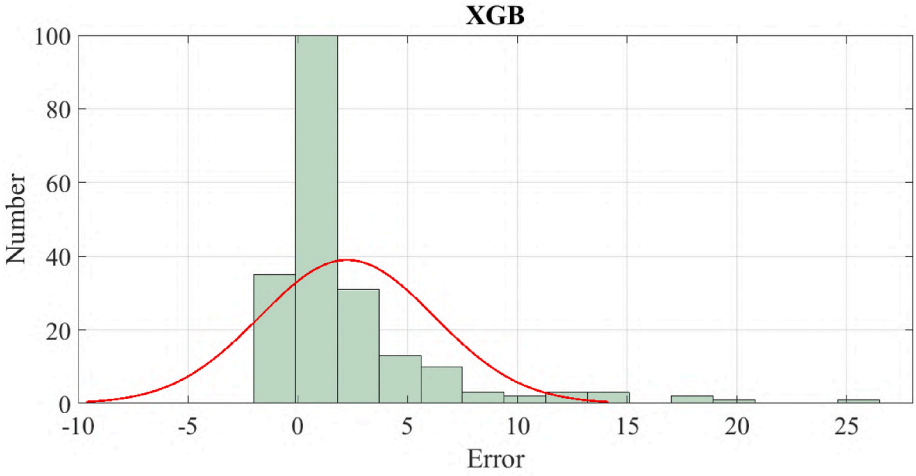


(r)

Fig. 4. (*continued*).



(s)



(t)

Fig. 4. (*continued*).

related to the MPR algorithm, and the highest error is related to the ELM algorithm. MAE and MARE indices are other indices that are close to each other, which according to two indices, RMSE and MSE, are related to the error of the points, and the smaller the error, the better. The MPR algorithm has the lowest error and the ELM algorithm has the highest error. The MAPE index is related to the error percentage of the points, and the lower the error percentage, the better the MPR algorithm with 7.5 % error, and the ELM algorithm with 221.76 % error, are the best and worst algorithms. According to other indicators, MBE is related to the error and according to the previous, the best value is related to the MPR algorithm, and the worst value is related to the ELM algorithm. Therefore, according to EC analysis, the best algorithm is MPR and the worst algorithm is ELM.

3.3. Results for AP

In this part, three parts of graphical analysis are discussed. The first part will deal with the graphical analysis of the error histogram. According to the previous explanations, the histogram chart is more optimal as long as its error column is close to zero. The error histogram diagram for all algorithms can be seen in Fig. 4. The error histogram for each of the twenty ML algorithms included in this research is shown in Fig. 4, which sheds light on how well they performed in terms of prediction. With a concentration of error values near zero being the ideal condition, the histogram shows the frequency distribution of prediction mistakes. To show the performance disparities, we have underlined the greatest error for each method. To illustrate its superior accuracy, the Multivariate Polynomial Regression (MPR) method displays the lowest maximum error among competing algorithms. The little dispersion of the error bars centering on zero in the MPR histogram shows that the experimental and projected viscosity values are quite close to one other. The Extreme Learning Machine (ELM) method, on the other hand, has the most error values, which are shown by a larger and more distributed histogram. This indicates that its predictions are very inaccurate. The histograms of the Adaptive Neuro-Fuzzy Inference System (ANFIS) and Gradient Descent (GD) algorithms reveal error distributions that are somewhat wider than MPRs but smaller than ELMs, indicating intermediate performance. Although it isn't quite as good as MPR, the Decision Tree (DT) method still shows a low error distribution, which means it's successful. Bayesian Network (BN), Broyden-Fletcher-Goldfarb-Shanno (BFGS), and Gaussian Process Regression (GPR) are three algorithms with histograms that have modest error distributions. This indicates that these algorithms strike a good compromise between computing efficiency and accuracy. Algorithms having a broader error

distribution, such as Partial Least Squares Regression (PLR) and Elastic Component Regression (ECR), are less effective at accurately predicting viscosity. With MPR showing the best results for viscosity prediction, and ELM and ECR the worst, the general trend in the histograms highlights the algorithmic performance variability. The dependability of the findings is directly affected by the distribution and quantity of mistakes; hence, the error histograms graphically show how important it is to pick the correct technique for predictive modeling. Consistent with the quantitative measures given in Table 1, our study confirms that MPR is a good fit for forecasting the viscosity of hybrid nanofluids. According to Fig. 4, the lower the maximum error, the more optimal the algorithm. For ANFIS, BFGS, BN, GD, LM, BPNN, DT, ECR, PLR, ELM, GMDH, GPR, Lasso, MLP, MLR, MPR, RBF, Ridge, SVM and XGB algorithms, the maximum error value is 23, 10, 23, 10, 9, 37, 9, 72, 50, 73, 13, 17, 63, 9, 50, 6.7, 11, 46, 53 and 25. According to Fig. 4, the lower the maximum error, the more optimal the algorithm. Therefore, the MPR algorithm has

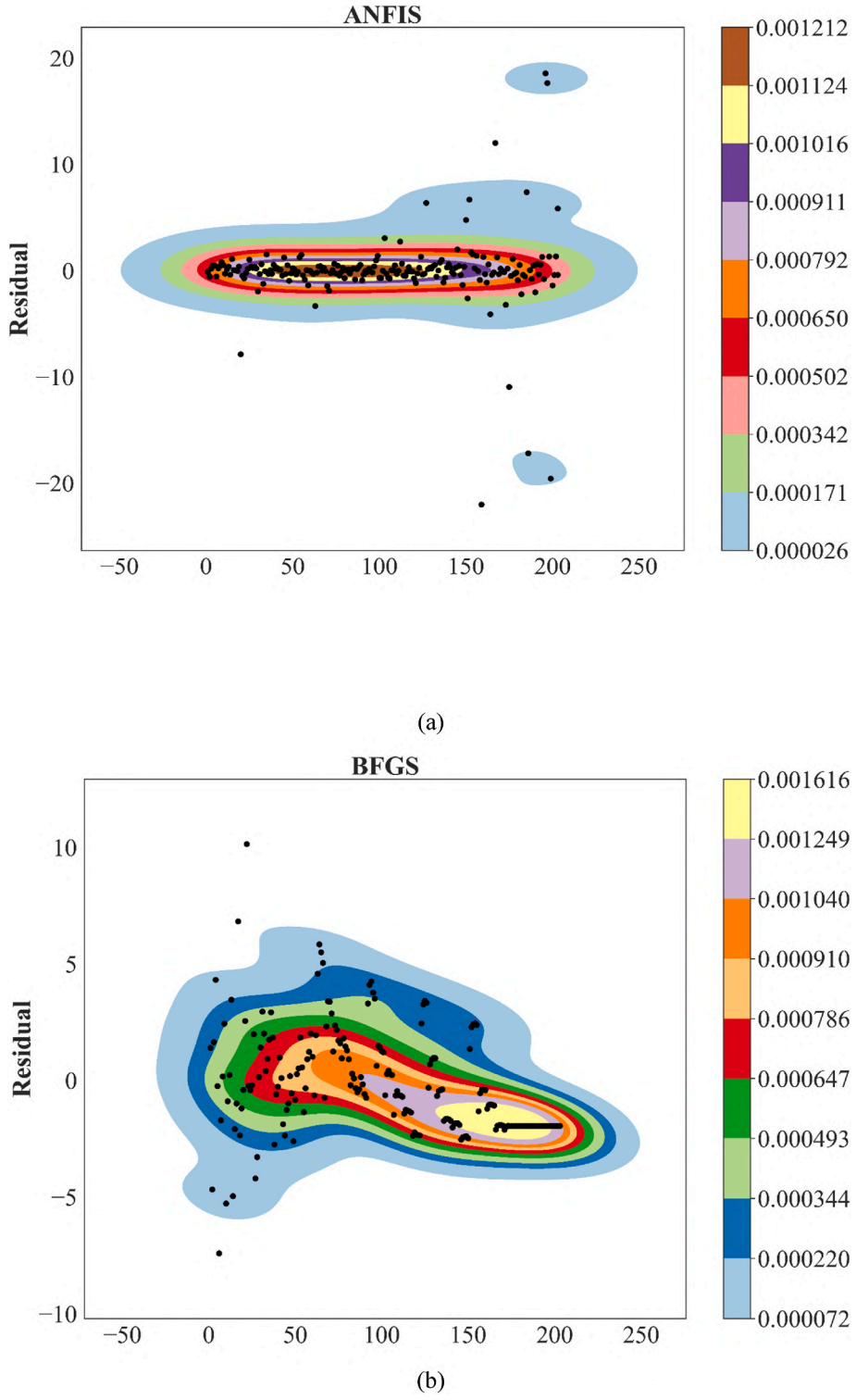
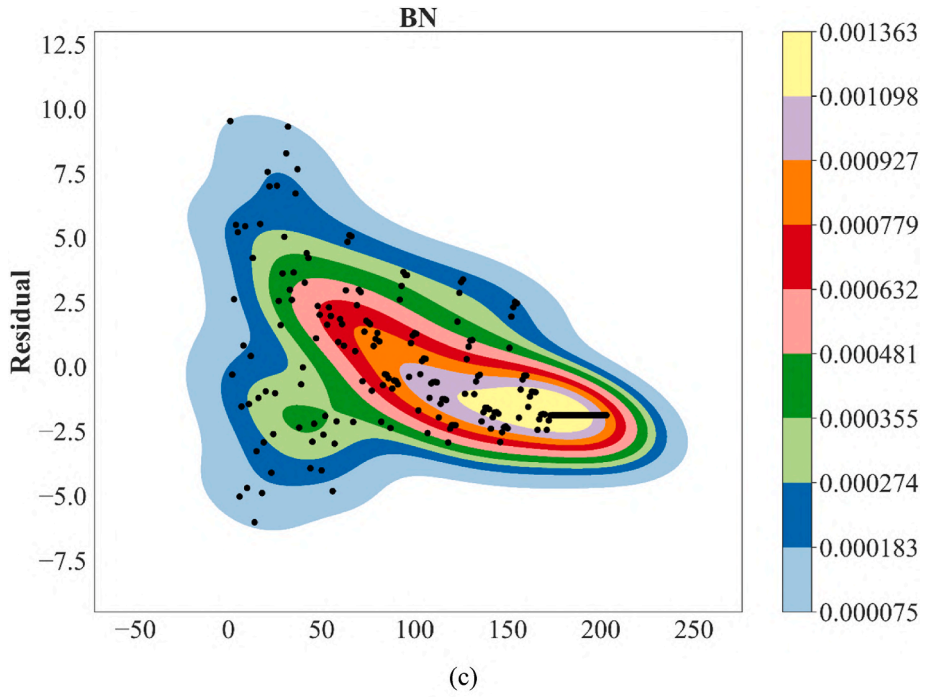
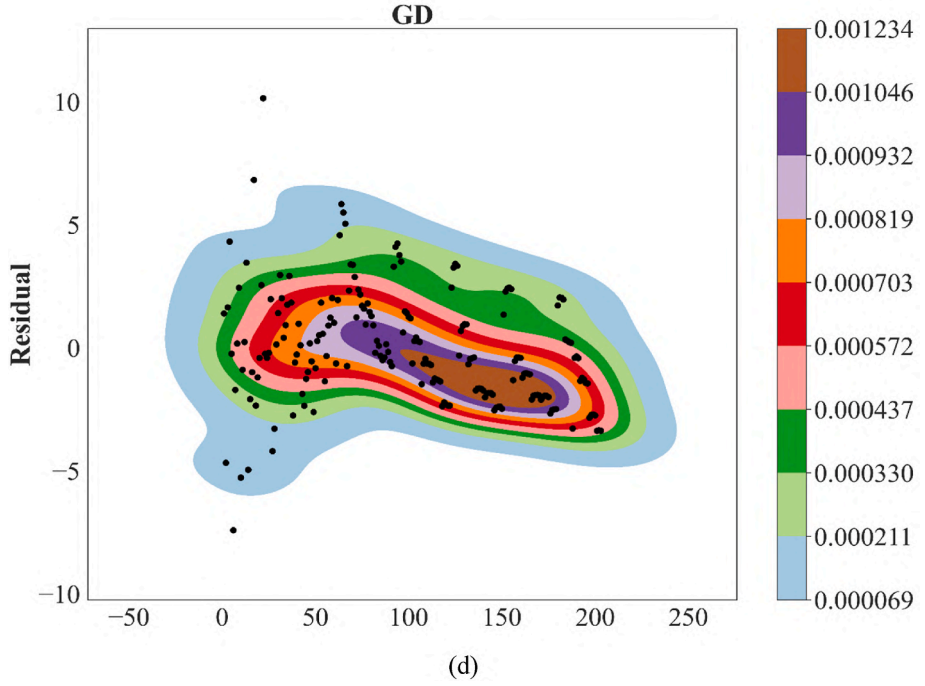


Fig. 5. Kde plot for a) ANFIS b) BFGS c) BN d) GD e) LM f) BPNN g) DT h) ECR i) PLR j) ELM k) GMDH l) GPR m) Lasso n) MLP o) MLR p) MPR q) RBF r) Ridge s) SVM t) XGB algorithm.



(c)



(d)

Fig. 5. (*continued*).

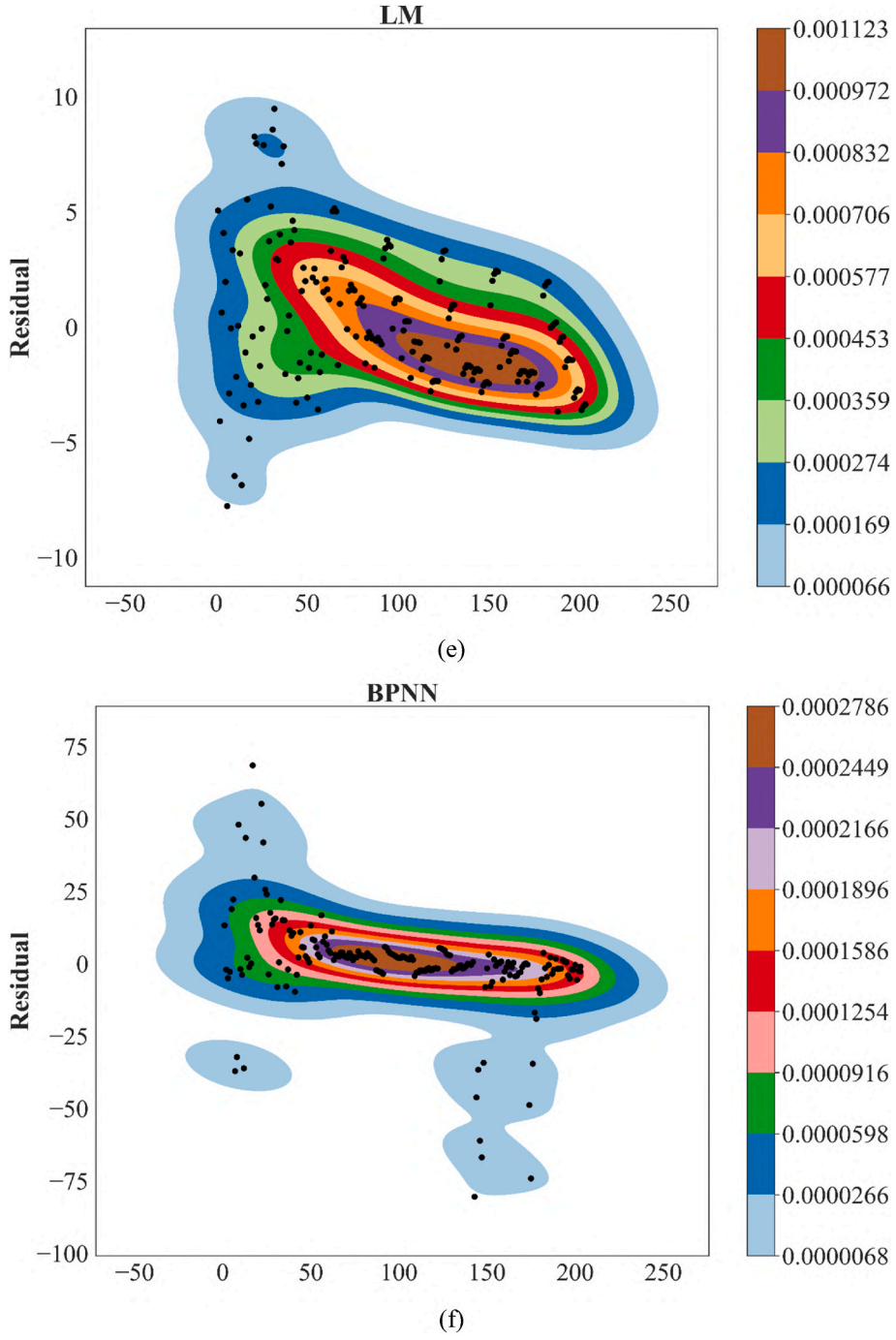


Fig. 5. (*continued*).

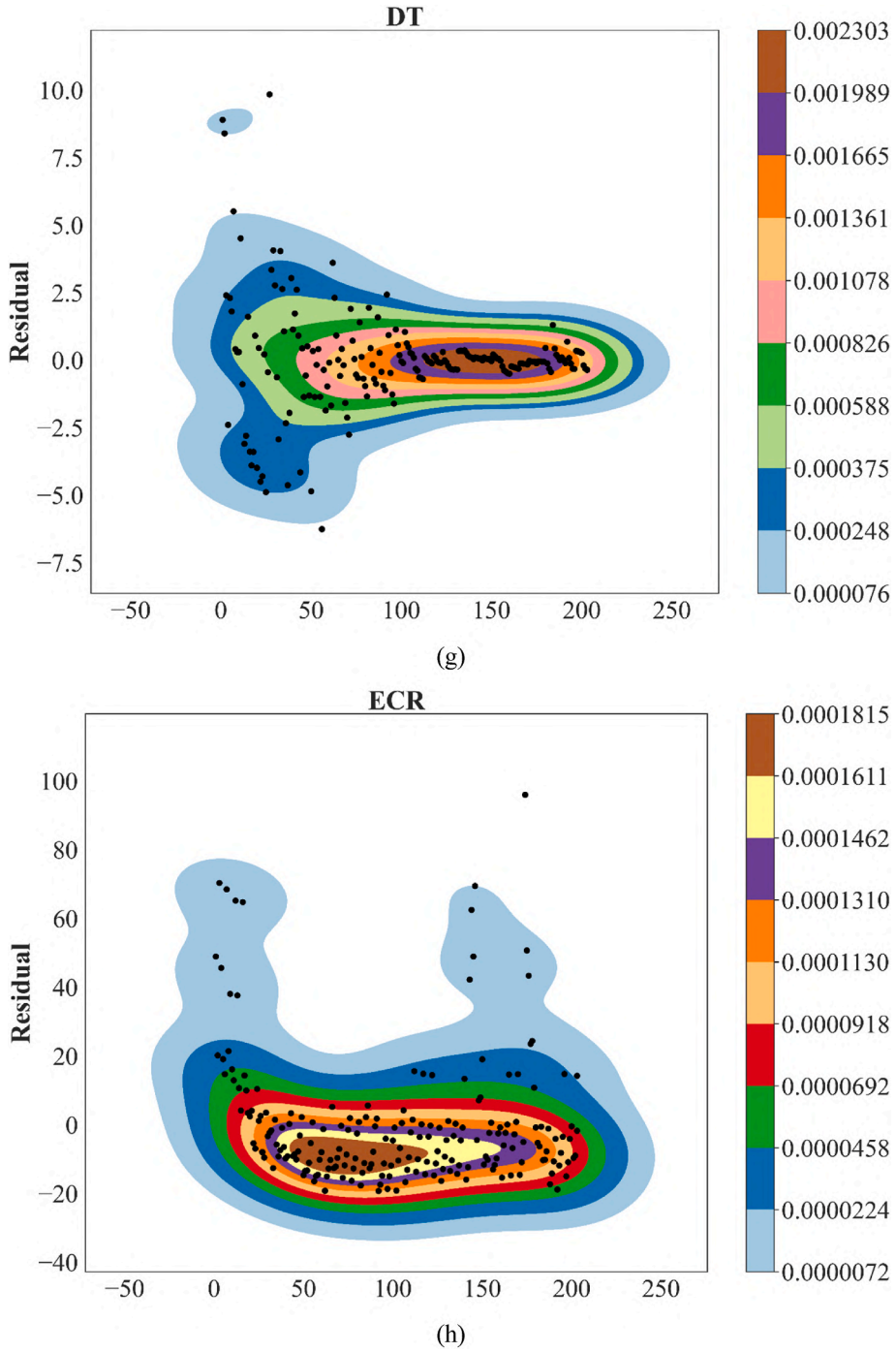


Fig. 5. (*continued*).

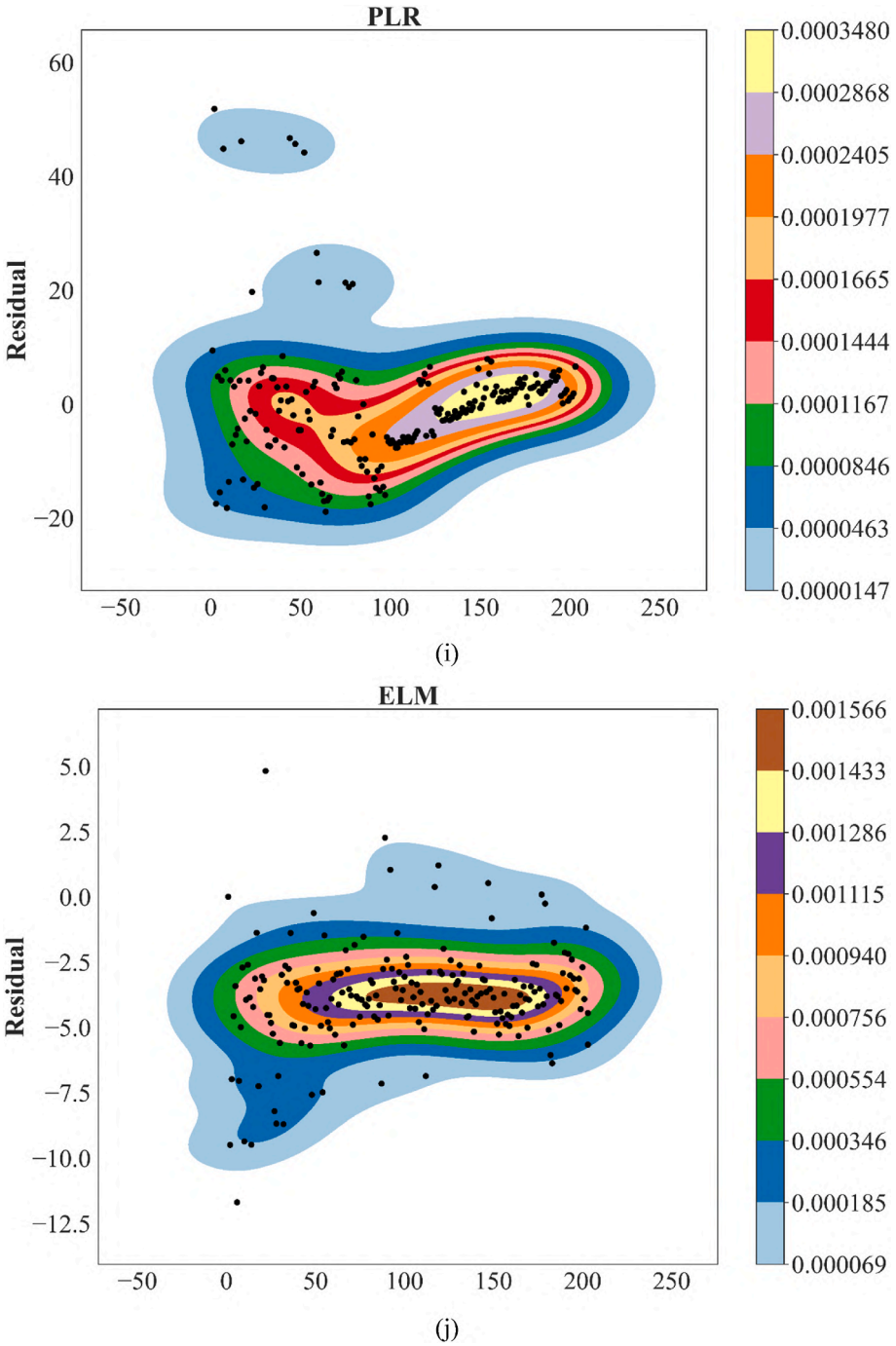
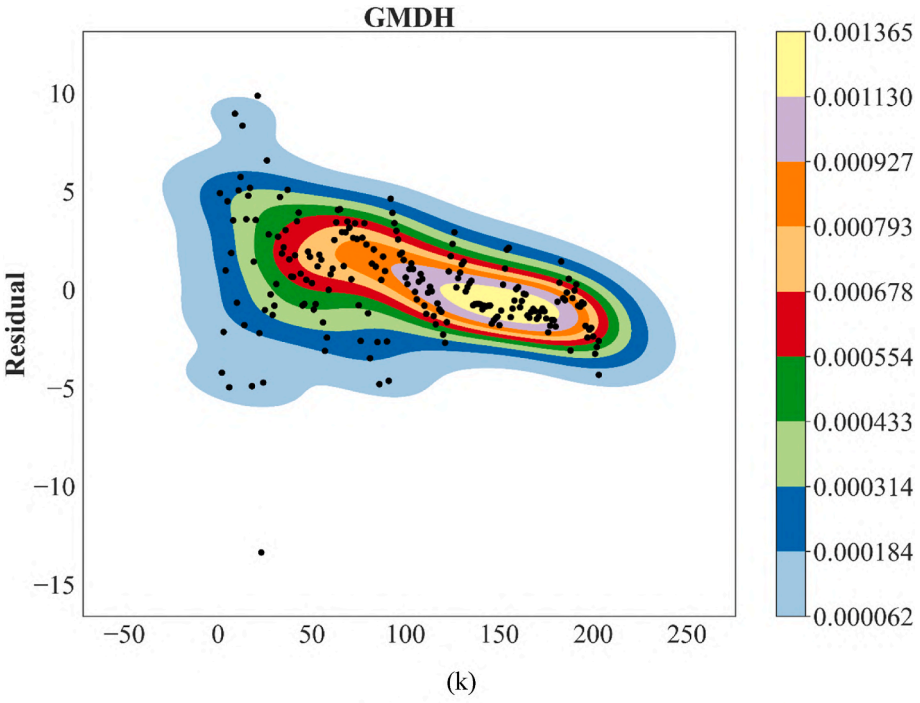


Fig. 5. (*continued*).



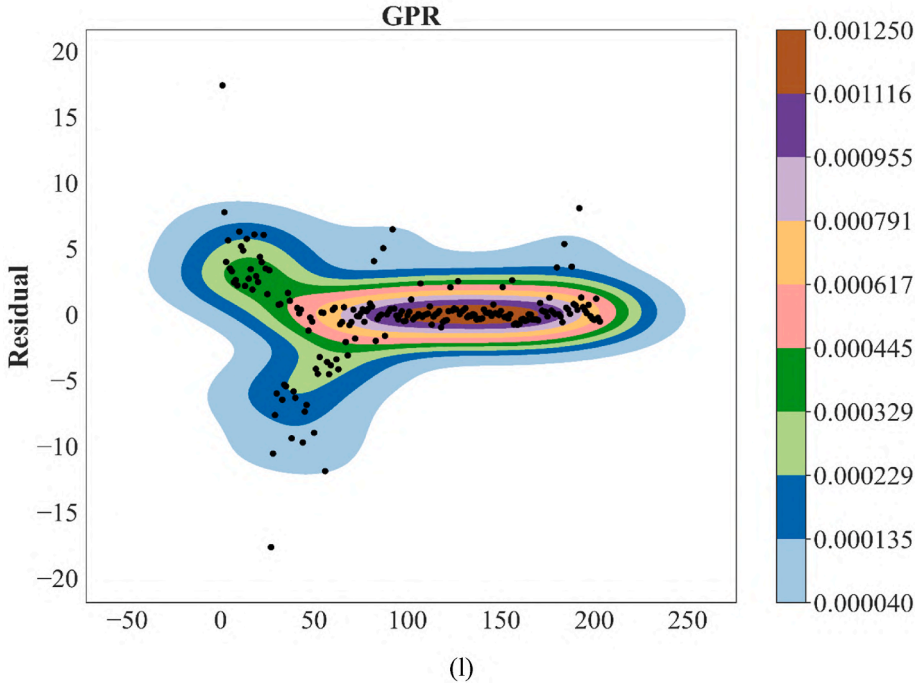


Fig. 5. (*continued*).

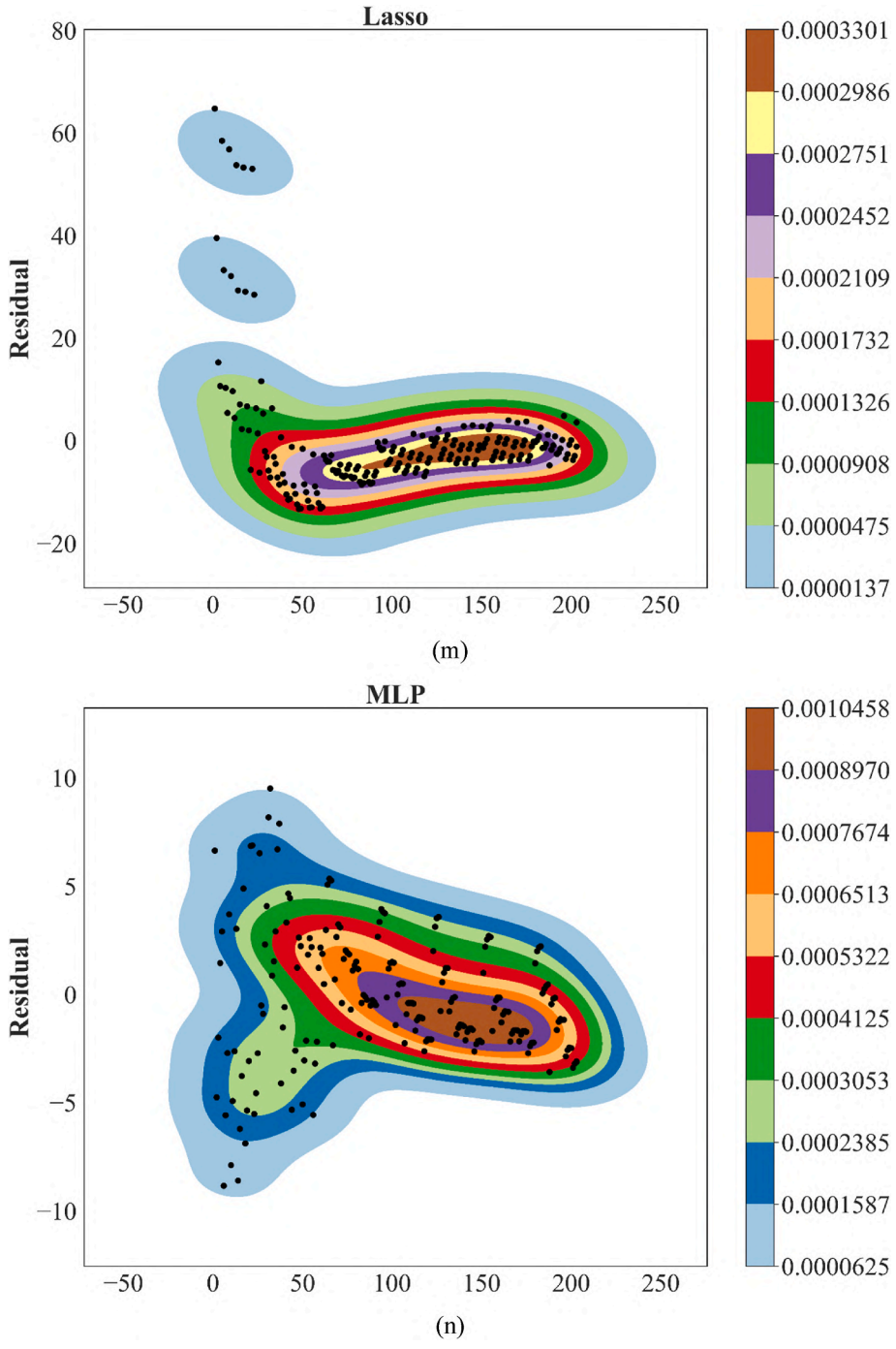


Fig. 5. (*continued*).

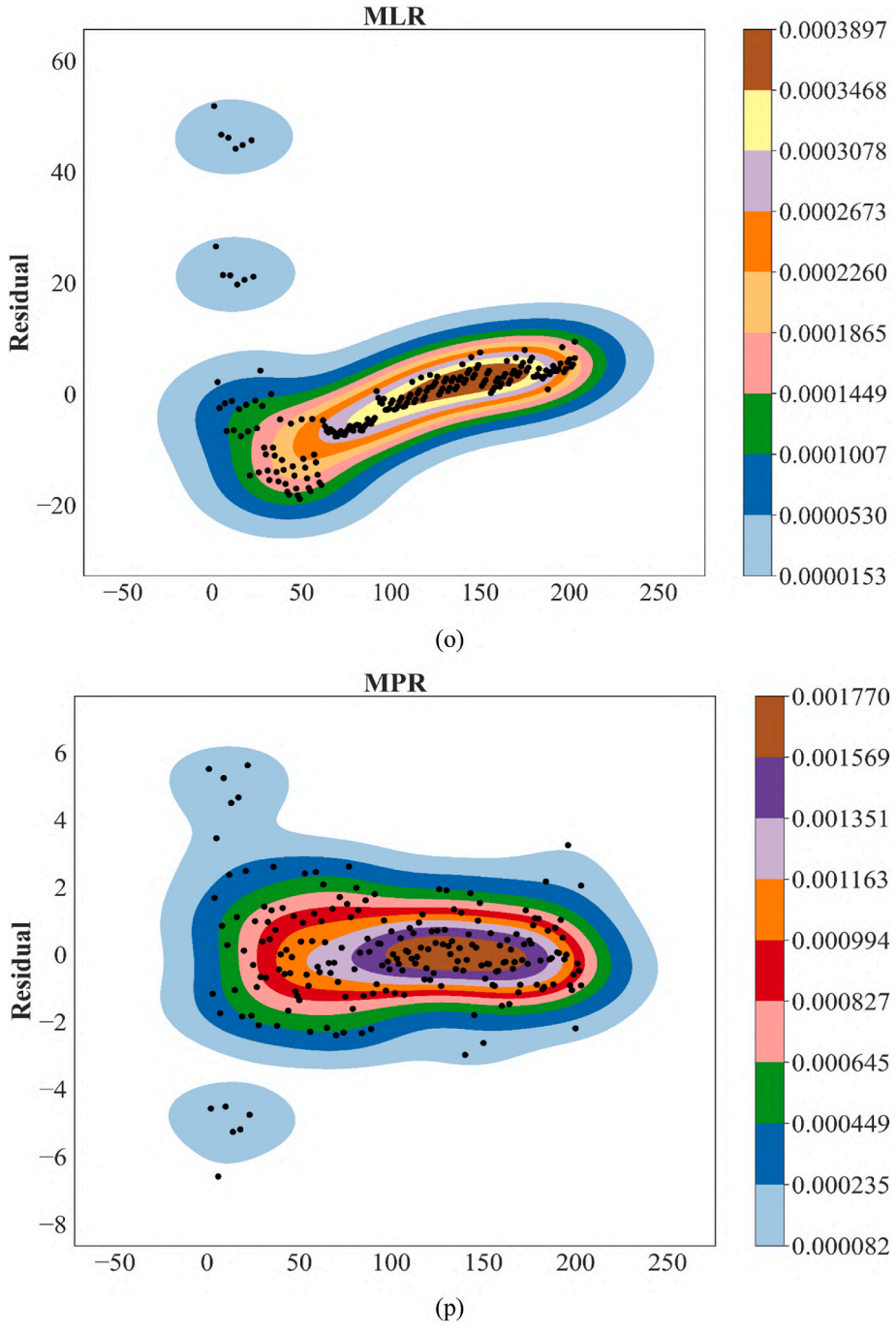


Fig. 5. (*continued*).

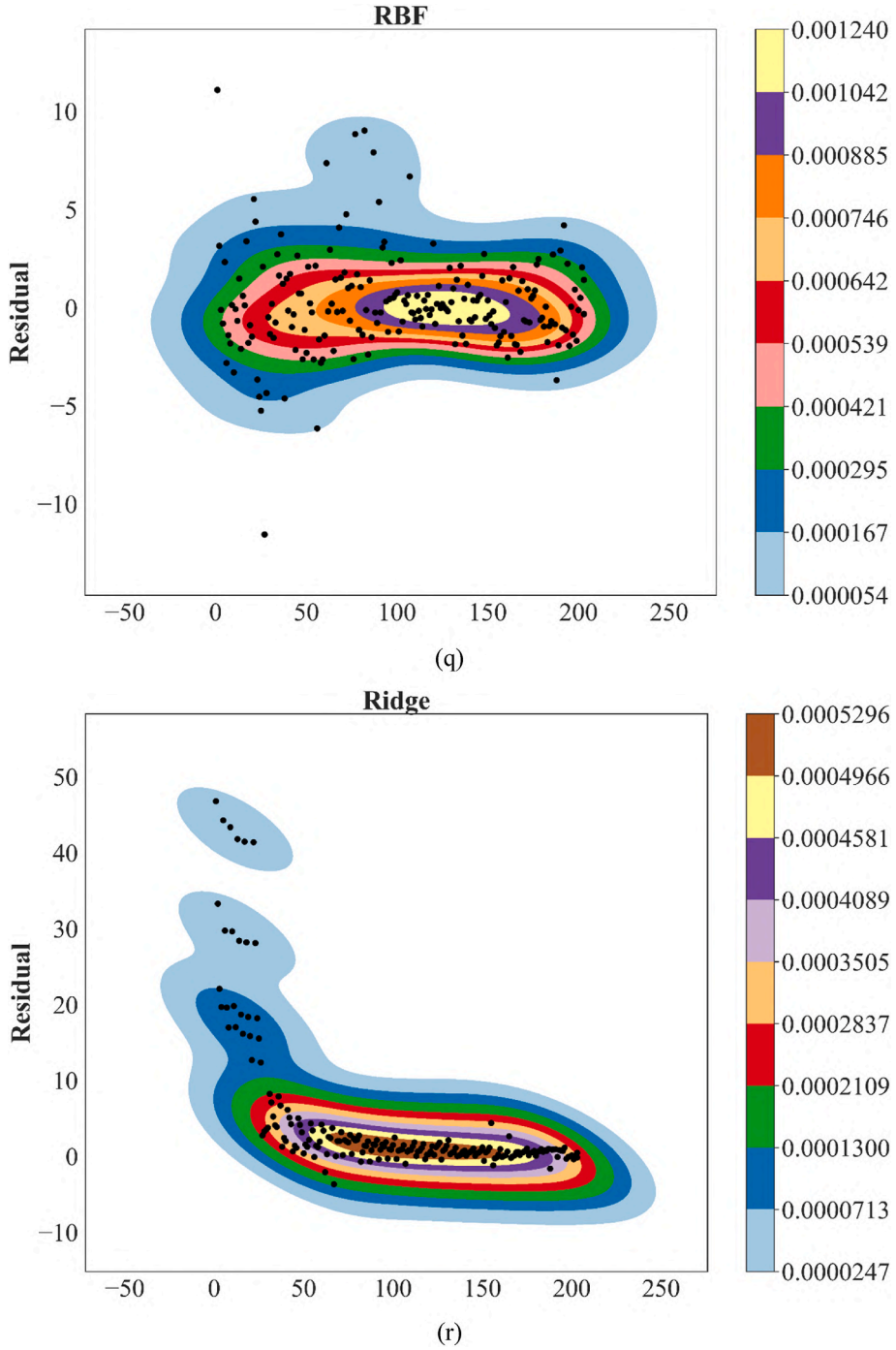
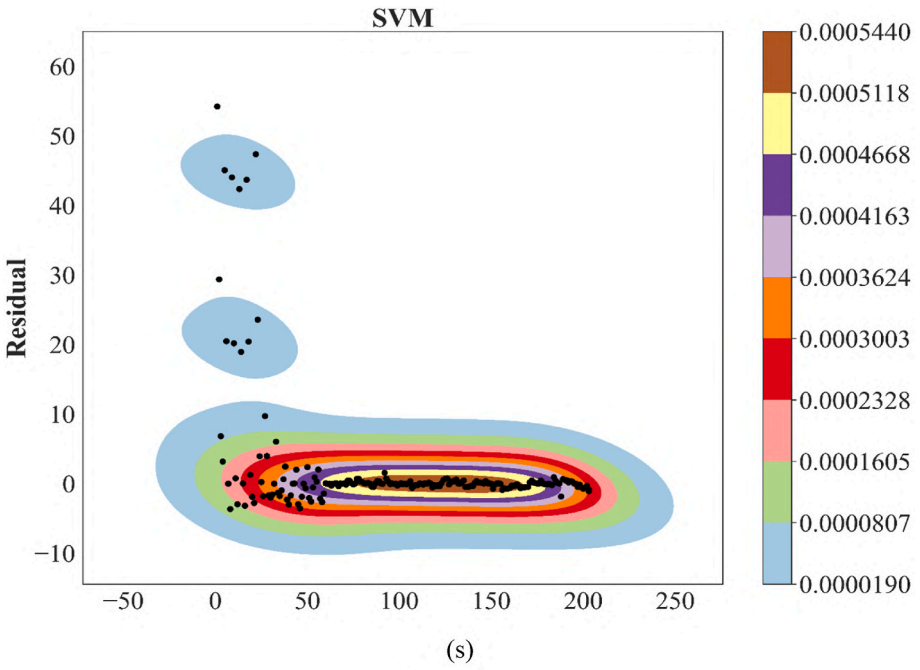
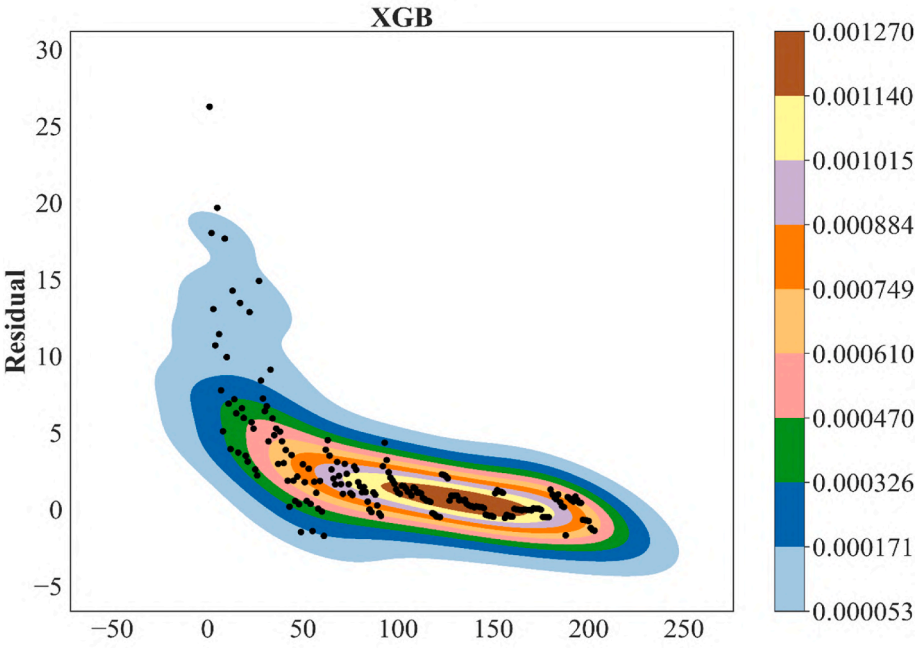


Fig. 5. (*continued*).



(s)



(t)

Fig. 5. (*continued*).

the lowest error and the ELM algorithm has the highest error. So, the best algorithm in terms of histogram is the MPR algorithm and the worst is the ELM algorithm.

The error histograms in Fig. 4 provide insights into the distribution of predicted errors across the SVM, RF, and MLP algorithms. The results indicate a close distribution of error by means of multivariate polynomial regression (MPR) centered at zero (i.e. when predicted and actual values deviate the least). While, the error distribution of ELM (Extreme Learning Machine) and ECR (Elastic Component Regression) is wide and scattered, which shows that predictions have more differences and the uncertainties of predictions. The better performing machine learning models actually capture more complex relationships between input and output. MPR is highly accurate, because the algorithm itself directly models the interaction between variables in polynomial equation form that are derived from the data. Methods like ANFIS (Adaptive Neuro-Fuzzy Inference System) and GPR (Gaussian Process Regression) also have good accuracy but with comparably lower than that of MPR because they can handle high-cost complicated nonlinear data owing to their versatility and ability to learn higher-dimension functions with the help of kernel. Physically, this means that we need to have a good understanding of the interaction between ϕ, T, and γ. MPR does a great job delivering this need with mathematical equations that reflect these interactions explicitly. On the computational side, random initialization of the weights leads to a sub-

Table 2
The performance of 20 machine learning algorithms in this study.

Algorithm	Peak Density (Near Zero)	Width of Error Distribution	Max Error Magnitude	Overall Accuracy
MPR	Highest	Narrowest	6.7	Excellent
ANFIS	High	Moderately Narrow	22	Good
GPR	High	Moderate	17.65	Good
DT	Moderate	Moderate	9	Good
BN	Moderate	Moderate	23	Average
BFGS	High	Moderately Narrow	10.15	Good
GD	Moderate	Moderate	10.15	Average
MLP	Moderate	Moderate	9.51	Average
BPNN	Moderate	Wide	37	Below Average
RBF	High	Moderately Narrow	11.55	Good
ECR	Lowest	Widest	96	Poor
PLR	Low	Wide	51.84	Poor
ELM	Lowest	Widest	−11.7	Poor
Lasso	Low	Wide	−64.74	Poor
XGB	Moderate	Moderately Narrow	25	Average
Ridge	Moderate	Wide	46.78	Below Average
SVM	Moderate	Wide	54.2	Below Average
GMDH	High	Moderate	−13.36	Good
LM	Moderate	Moderate	9.47	Good
ECR	Lowest	Widest	96	Poor

optimal solution, for example, in algorithms like the extreme learning machine (ELM). In contrast, MPR solved this issue and achieved much more precise prediction due to its derivation of constant equations that were tuned directly to the data set.

Among other analysis charts, there is a Kde chart with error values indicated. According to this chart, the number of points close to zero indicates that the chart has less error and is a better algorithm. The error value is minimal for an algorithm with proper correlation. These values can be seen in Fig. 5. Fig. 5 shows the error value kernel density estimation (KDE) plots for twenty machine learning algorithms. The explanation for this figure emphasizes the prediction accuracy and dependability of the techniques. For each method, these KDE graphs show the probability density of mistakes; a peak close to zero indicates little prediction error. The KDE plot of the Multivariate Polynomial Regression (MPR) method shows a tiny peak near zero, further demonstrating its excellent performance. This shows that it is quite accurate and reliable, as most of its predictions match the observed viscosity values very well. However, the Extreme Learning Machine (ELM) algorithm's prediction performance and considerable departures from the experimental data are shown by its wide and scattered KDE plot, which has a peak that is far from zero and a significant error density at larger magnitudes. As their KDE plots are modestly concentrated around zero but with larger bases compared to MPR, algorithms like Adaptive Neuro-Fuzzy Inference System (ANFIS), Gradient Descent (GD), and Gaussian Process Regression (GPR) show moderate performance. Predictions made by these algorithms are not as accurate as MPRs, but they do a decent job of capturing patterns in viscosity data. Bayesian Network (BN) and Decision Tree (DT) algorithms also do a respectable job; KDE plots reveal acceptable peaks around zero, albeit error variability is somewhat larger. Alternatively, KDE graphs generated by Elastic Component Regression (ECR) and Partial Least Squares Regression (PLR) are distorted and uneven, indicating poor accuracy and large predicted errors. Particularly inaccurate is the ECR KDE figure; the error density there is very non-zero and spans a wide range. In comparison to the best algorithms, the Radial Basis Function (RBF), Multi-Layer Perceptron (MLP), and Backpropagation Neural Network (BPNN) algorithms display intermediate behavior; their KDE plots show peaks close

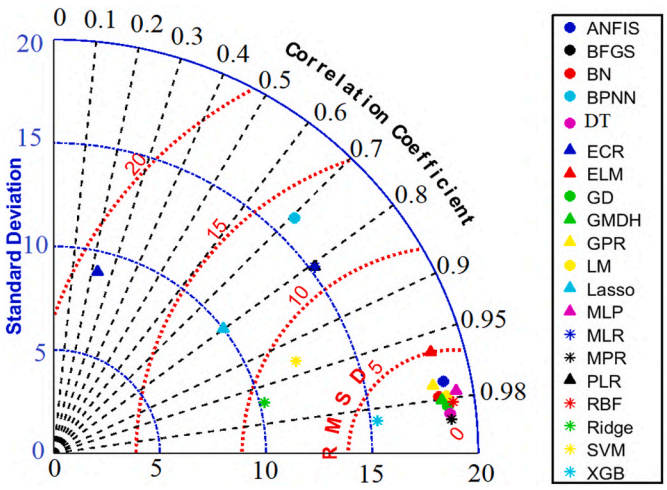


Fig. 6. Taylor diagram for 20 MLA

to zero but with wider dispersion. These trends indicate that these algorithms can make educated guesses, but they can't compare to MPR in terms of accuracy and reliability. The wide variation in the error distributions across the algorithms is shown by the overall analysis in Fig. 5. SVF, temperature, and shear rate are the input factors, and MPR's narrow and clearly defined KDE plot demonstrates its ability to accurately represent the nonlinear interactions between these variables and the output variable, viscosity. On the other side, ELM and ECR's scattered KDE plots show that they do a poor job of capturing the data's complicated dynamics. Since MPR consistently underperformed in this investigation, it is clear that using the right algorithm is crucial for making accurate and dependable viscosity predictions. According to Fig. 5, the lower the error rate, the more optimal the algorithm. For ANFIS, BFGS, BN, GD, LM, BPNN, DT, ECR, PLR, ELM, GMDH, GPR, Lasso, MLP, MLR, MPR, RBF, Ridge, SVM, and XGB algorithms, the maximum error value is −22, 10.15, 9.53, 10.15, 9.47, −80.48, 9.84, 96, 51.84, −11.7, −13.36, −17.65, −64.74, 9.51, 51.84, −6.59, −11.55, 46.78, 54.2 and 26.27, respectively. Therefore, the MPR algorithm has the least error and the ECR algorithm has the most error. Therefore, the best algorithm in terms of the correlation diagram is the MPR algorithm and the worst is the ECR algorithm. Table 2 shows the performance of each algorithm.

Another analysis diagram is the Taylor diagram. In this diagram,

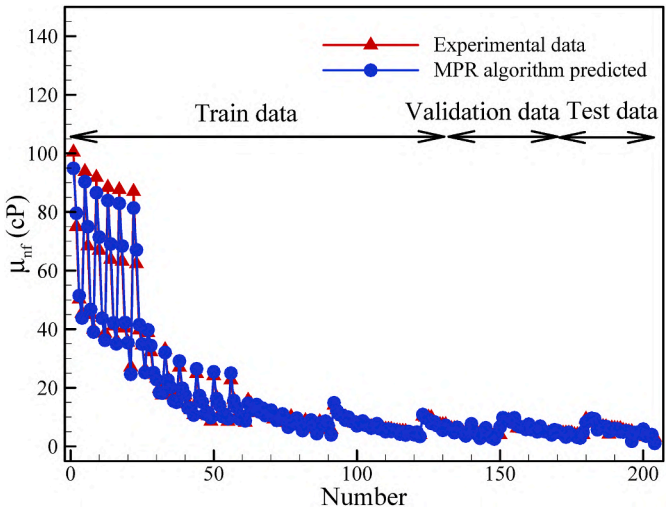


Fig. 7. Plot of experimental data with the points predicted by the MPR algorithm.

three indicators R, STD and RMSD are specified. Taylor's diagram can be seen in Fig. 6. For ANFIS, BFGS, BN, GD, LM, BPNN, DT, ECR, PLR, ELM, GMDH, GPR, Lasso, MLP, MLR, MPR, RBF, Ridge, SVM and XGB algorithms, the value of R is equal to 0.9827, 0.9927, 0.9888, 0.9927, 0.9889, 0.7054, 0.9948, 0.2299, 0.8069, 0.9641, 0.9903, 0.9839, 0.7996, 0.9877, 0.8069, 0.9961, 0.9913, 0.9708, 0.931 and 0.9946. For ANFIS, BFGS, BN, GD, LM, BPNN, DT, ECR, PLR, ELM, GMDH, GPR, Lasso, MLP, MLR, MPR, RBF, Ridge, SVM and XGB algorithms, the value of RMSD is equal to 3.4975, 2.2826, 2.8378, 2.2826, 2.8085, 13.66, 1.9187, 18.955, 11.1533, 5.02, 2.6335, 3.4022, 12.4295, 3.0034, 11.1533, 1.6593, 2.4972, 9.2743, 8.7171 and 3.9572. For ANFIS, BFGS, BN, GD, LM, BPNN, DT, ECR, PLR, ELM, GMDH, GPR, Lasso, MLP, MLR, MPR, RBF, Ridge, SVM and XGB algorithms, the value of STD is equal to 18.6925, 18.7, 15.3369, 18.7, 18.71, 16.054, 18.783, 8.995, 15.234, 18.4112, 18.4683, 18.1558, 10.01, 19.189, 15.234, 18.807, 18.997, 10.236, 12.241 and 15.337. Therefore, the best algorithm in terms of the correlation diagram is the MPR algorithm and the worst is the ELM algorithm.

One of the most versatile implicit functions providing a plot of multiple values of statistical metrics (correlation coefficient (R), standard deviation (STD), root mean square error (RMSD), etc.) is the Taylor plot. Furthermore, Taylor plots are useful to assess and compare the predictive performances of machine learning models in predicting complex physical properties, e.g. viscosity. Overall, the significance of this plot is that it gives a condensed general overview of how well the model predicts, how much it varies around that prediction, and what systematic bias may exist. In this plot, the correlation coefficient (R) is the radial distance to each point from the origin. And a reading of value in a range of 1 reflects a strong relationship between the predicted values and actual values. As shown in Fig. 6, the MPR algorithm (R = 0.9961) is more effective than ECR (R = 0.2299) and other algorithms. An STD represents how much variability there is between the predicted and the actual values. The nearer points to the reference point (empirical values) are a more significant fit of the model to the data variations. The value of the standard deviation of MPR (18.807) in the Taylor plot is nearly identical to the true value, while ELM and ECR are far from the reference value, indicating little ability to reproduce the variation of the data. The root mean square error (RMSD) describes the mean deviation of the predictions, where lower values indicate more accuracy of the model. The error is lowest in MPR with RMSD = 1.6593, whereas ELM (5.02) and ECR (18.955) present a much larger error in this comparison. From a geometric relationship perspective, in a Taylor plot, the closer the models' points are to the reference point, the better their performance, given that proximity signifies high correlation, low root mean square deviation (RMSD), and STD around a true value. The distance of the model point from the reference point shows its better performance, and the position of the MPR close to the reference point justifies its ability of viscosity prediction. Models with high STD or RMSD are inaccurate or over/under-predicting viscosity. For instance, a significant distance of the ECR from the reference point suggests inherent issues with the accuracy and variability of its perturbative predictions. Some models might outperform (2) on one criterion, but fail on another. Specifically, ANFIS reported R = 0.9827 but with a larger error (RMSD = 3.4975) compared to MPR, which shows the trade-off between the accuracy of the correlation and the amount of error. This Taylor plot sees which models learn complex and nonlinear relationships. The powerful performance of the MPR demonstrates that the proposed model is a good way to model the complex interactions of input factors, such as volume fraction of nanoparticles, temperature, and shear rate and viscosity. Models like ELM and ECR that have low R, high RMSD, and disproportionate STD are bad choices and easily spotted thus emphasizing a key issue about model choice.

According to the analysis of EC and AP, the best algorithm is the MPR algorithm. The next goal is to extract the mathematical equations between the inputs and the output and plot the experimental data along with the data predicted by the MPR algorithm.

3.4. Derivation of mathematical equations between inputs and outputs

Machine learning models can address nonlinear viscosity behavior by capturing complex relationships between the input (nanoparticle volume fraction, temperature, shear rate) and output variables (viscosity). MPR directly captures nonlinear relationships (such as γ^2 and T^3) and interactions between independent variables (e.g., T·γ) and can closely define the properties of the hybrid nanofluid's complex viscosity. Neural Networks like MLP and BPNN approximate non-linear mappings from input to output via a hidden layer or to avoid overfitting they need to be tuned carefully. Decision tree-based models (ex, DT, and XGBoost) construct local trends by recursively partitioning the data into partial subsets and then modeling the nonlinear relationship. e.g. Gaussian regression (GPR) and support vector machine (SVM): These methods take advantage of transformations by translating the data into higher-dimensional spaces, which lead to nonlinear relationships highlighted by linear dependencies, thus providing very good approximations of the viscosity as a function of input variables. This also promotes the use of adaptive systems like ANFIS, which merges fuzzy and neural networks to maximize flexibility in modeling the nonlinear nature and uncertainties in nanofluid behavior. These methods enable nonlinear trends in viscosity prediction to be modeled with high accuracy. Of these, the multivariate polynomial regression (MPR) method has been identified as the most efficient model as regressions express polynomial interactions explicitly.

After it is determined that the MPR algorithm is superior to other algorithms, it is possible to extract the mathematical relationships between its inputs and outputs. Equation (42) shows this mathematical relationship.

$$\mu_{nf} = 542.28 + 2.370462*\gamma+0.059*\gamma^2-1.9*e^{-3}*\gamma^3+2.44*e^{-5}*\gamma^4-72.24*T-0.37*T*\gamma+2.2*e^{-4}*T*\gamma^2+5.3*e^{-6}T*\gamma^3+7.12*e^{-8}*T*\gamma^4+3.94*T^2+0.02*T^2*\gamma-1.32*e^{-5}*T^2*\gamma^2-2.55*e^{-7}*T^2*\gamma^3-0.11*T^3-3.1*e^{-4}*T^3*\gamma+4.05*e^{-7}*T^3*\gamma^2+1.47*e^{-3}*T^4+2.02*e^{-6}*T^4*\gamma-188.01*\varphi-3.44*\varphi*\gamma+0.1*\varphi*\gamma^2-1.3*e^{-3}*\varphi*\gamma^3+6.58*e^{-6}*\varphi*\gamma^4+26.37*\varphi*T-0.19*\varphi*T*\gamma+5.7*e^{-4}*\varphi*T*\gamma^2+3.51*e^{-6}*\varphi*T*\gamma^3-0.88*\varphi*T^2+5.52*e^{-3}*\varphi*T^2*\gamma-1.3*e^{-5}*\varphi*T^2*\gamma^2+1.11*e^{-2}*\varphi*T^3-4.74*e^{-5}*\varphi*T^3*\gamma-4.24*e^{-5}*\varphi*T^4-205.36*\varphi^2+15.98*\varphi^2*\gamma-0.11*\varphi^2*\gamma^2-3.88*e^{-4}*\varphi^2*\gamma^3-6.86*\varphi^2*T-0.06*\varphi^2*T*\gamma-7.04*e^{-5}*\varphi^2*T*\gamma^2+0.28*\varphi^2*T^2+6.45*e^{-4}*\varphi^2*T^2*\gamma-2.82*e^{-3}*\varphi^2*T^3+367.62*\varphi^3-17.57*\varphi^3*\gamma+0.11*\varphi^3*\gamma^2-3.73*\varphi^3*T+0.01*\varphi^3*T*\gamma+0.02*\varphi^3*T^2-95.55*\varphi^4+4.58*\varphi^4*\gamma+0.85*\varphi^4*T-5.43*\varphi^5-8.05*e^{-6}*T^5-1.23*e^{-7}*\gamma^5 \quad (42)$$

It is also possible to draw a graph of the predicted points next to the experimental points, which is according to Fig. 7. According to this diagram, it is clear that the MPR algorithm has a very high performance and the predicted points are very close to the experimental points and have a small error. In Fig. 7, we can see how the experimental data points and the anticipated points from the MPR algorithm are visually compared. The figure's value is in the way it graphically shows how well the MPR algorithm predicts the non-Newtonian hybrid nanofluid's viscosity (μ_{nf}) from data collected using iron, copper, water, and ethylene glycol. The prediction points are more accurate when they are closer to the experimental points, as shown in Fig. 7, which compares the actual viscosity values with the predicted viscosity values generated by the MPR algorithm. The MPR algorithm accurately predicts the nanofluid viscosity, as seen in Fig. 7, where the predicted points are very congruent with the observed sites. The ability to visually compare expected and experimental results gives a clear picture of how well the model is doing, which is a crucial confirmation of the model's correctness. Fig. 7's tightly packed dots demonstrate that the MPR algorithm achieves a low prediction error, in line with the quantitative error metrics (RMSE and MAE) mentioned in Table 1.

From Fig. 7, we can observe a good agreement between the experimental viscosity (μ_{nf}) values and those predicted by the MPR algorithm. The diagonal distribution of the points implies an excellent agreement between the experimental and predicted values. The low scatter of data indicates little prediction error, confirming the high performance of MPR. MPR models the system behavior with great accuracy since it exploits the data structure and its physical foundations making it outperform its competitors. Although other algorithms can perform adequately, some are unable to meet the level of accuracy due to slight biases or poor variance. Physically, the mathematical equation resulting from MPR describes the fundamental laws of nanofluids' viscosity with high accuracy. From a computational perspective, MPR avoids overfitting by avoiding excessive complexity while allowing just about the right amount of freedom to model the specifics of the data set. ECR (Elastic Component Regression) and PLR (Partial Least Squares Regression) algorithms suffer more errors relying on a simplification of the problem.

4. Conclusion

This study systematically developed twenty machine-learning algorithms to predict the viscosity of non-Newtonian iron-CuO/water-ethylene glycol hybrid nanofluids. The impact of the solid volume fraction (SVF), temperature, and shear rate on viscosity was studied, and it was found that SVF is the most dominant parameter affecting viscosity, followed by shear rate and temperature. Of the tested algorithms, Multivariate Polynomial Regression (MPR) yielded the highest correlation coefficient ($R2 = 0.992$) and lowest error metrics ($RMSE = 1.66$, $MAE = 1.146$). In terms of accuracy, however, although the Extreme Learning Machine (ELM) algorithm fared poorly, this serves to reinforce that appropriate algorithms must be selected for tasks such as these. The main contribution of this study is the mathematical equation derived from MPR, to express the relationship between input variables and viscosity in a few words. This is used for engineers and researchers to be able to find the viscosity of a fluid without computational models. The analysis and interpretation of model performance and results required the use of advanced data visualization techniques, including heatmaps, box plots, error histograms, kernel density estimation (KDE) plots and Taylor diagrams. While it was invaluable to see the strengths and weaknesses of each particular algorithm, it was even more valuable to realize the features of the tools available for the complete evaluation of predictive models. Accurate prediction of viscosity affects all parameters such as pumping power, Reynolds number, and heat transfer coefficients used when optimizing heat transferring systems. Though this study targeted iron-CuO/water-ethylene glycol hybrid nanofluids, the methodologies developed here are applicable to be employed in other nanofluid systems. In future studies, more features like nanoparticle size, shape, and surface chemistry may be included, and more sophisticated approaches, such as ensemble methods and deep learning, may be used to achieve better prediction accuracy. In summary, the current paper demonstrates the suitability of MPR (machine learning) as a tool for predicting the viscosity of non-Newtonian hybrid nanofluids which could have significant theoretical and practical engineering system implications.

5. Limitation and future direction

The analysis performed in this study only targets iron-CuO/water-ethylene glycol hybrid nanofluids, so it might not be extended to other incipient nanofluids or base fluids. Nanofluids with different nanoparticle materials (e.g., graphene oxide, silver) or even completely different base fluids (e.g., oil-based systems) may have different behavior. Generally, y is called input parameters in the study, the study takes only three input parameters, solid volume fraction (SVF), temperature and shear rate. The model does not account for factors like nanoparticle size, shape, surface chemistry, and aggregation effects1 that could affect viscosity predictions under specific conditions. Range of Conditions Although the experimental data cover a relatively narrow range of SVF, temperature, and shear rate values, the model is not able to make predictions outside the ranges of the training data. If Multivariate Polynomial Regression (MPR) is modeled very successfully within fitted limits, the model might tend to lose its ability to capture non-linear trends at the tails, where complex interactions have dominance. At very high temperatures for example, the rheological properties of the nanofluid may become erratic due to thermal degradation or phase changes. At sufficiently high shear rates, shear thinning may enter more complex regimes that MPR cannot do justice to. While MPR does not grapple with polynomial overfitting like polynomial regression, there can be an overfitting risk if the underlying relationships are particularly complex and/or noisy outside the training data range.

Since important factors like pumping power, Reynolds number, and heat transfer coefficients are directly affected by viscosity, it is crucial to accurately anticipate viscosity to optimize heat transfer systems. To help engineers create coolers and heat exchangers that work better, the MPR model provides a trustworthy and effective technique for predicting viscosity. The simplified prediction method made possible by the developed mathematical equation also makes it applicable to real-world scenarios. There are a few caveats to be aware of, even if this work does provide some great improvements in viscosity prediction. The research only looked at one kind of hybrid nanofluid—iron-CuO/water-ethylene glycol—so the findings may not apply to other types of nanofluids. The SVF, temperature, and shear rate were the only variables that could be entered. To improve the accuracy of predictions, future research may take into account other characteristics including the size, shape, and surface chemistry of nanoparticles. Improving prediction accuracy might be a future research focus by investigating advanced machine learning approaches like ensemble methods and deep learning. To confirm the MPR model's applicability, the research should be expanded to include a broader variety of nanofluid compositions and operating circumstances. One way to make the MPR model more useful in industrial heat transfer applications is to include it in control and monitoring systems that work in real-time. The paper highlights the possibility of using machine learning methods, namely Multivariate Polynomial Regression, to forecast the viscosity of hybrid nanofluids that do not adhere to Newton's laws. This study helps improve heat transfer technology by developing a strong mathematical model and determining the main components that affect viscosity. The results provide valuable insights into the behavior of nanofluids and provide practical methods for improving engineering systems, which in turn leads to better energy efficiency and overall performance. By utilizing deep learning models like Convolutional (RNNs) or CNNs, it can capture those relatively complex higher-order interactions and temporal dependencies which might enhance the prediction performance. These models work best for large data sets containing complex patterns. Challenges: Since deep learning requires a lot of data and time to train molds, it does If one does not apply data correctly, it may be said to be unnecessary. Larger models with additional layers might not generalize effectively given the small amount of available data (204 samples), so MPR might train sufficiently and will outcompete more complex methods (deep learning approaches) for this amount of data. Better Predictive Power: A model trained on another but closely related dataset can achieve better performance on small datasets (transfer learning). As an example, for a prediction of the viscosity of mono-nanofluids, the same pre-trained neural network can be fine-tuned for hybrid nanofluids. This tactic could enable more precise techniques while not asking for vast quantities of additional data. Transfer learning would be especially beneficial in cases where one would need to extend the model to new combinations of nanofluid compositions or new operating regimes. The mathematical formula obtained from the experimental results from MPR was verified and confirmed for use in practical real-time control for heat exchangers, cooling systems, and other industrial applications. This enables engineers to change the parameters (flow rate, temperature, etc.)

dynamically depending on predicted viscosity values. It is vital to validate over a range of operational scenarios to deploy this model in wide-scale industrial use. This involves testing at different flow regimes, pressure levels, and contamination effects. Integrating into Monitoring Systems: You could integrate the model into monitoring systems with feedback loops to continuously optimize system performance. For instance, you could set alerts or automatically adjust conditions in real time based on viscosity predictions to manage the process for optimal operation. Answers could refer to historical data recorded during operation, which could retrain that model.

CRediT authorship contribution statement

Mohammed Shorbaz Graish: Funding acquisition, Software, Formal analysis, Writing – original draft, Writing – review & editing. **Ali B.M. Ali:** Funding acquisition, Software, Formal analysis, Writing – original draft, Writing – review & editing. **Murtadha M. Al-Zahiwat:** Funding acquisition, Software; Formal analysis, Writing – original draft, Writing – review & editing. **Saja Mohsen Alardhi:** Writing – review & editing, Conceptualization, Data curation, Formal analysis, Supervision, Investigation, Writing – original draft, Writing – review and editing. **Mohammadreza Baghoolizadeh:** Writing – review & editing, Conceptualization, Data curation, Formal analysis, Supervision, Investigation, Writing – original draft, Writing – review & editing. **Soheil Salahshour:** Funding acquisition, Software, Formal analysis, Writing – original draft, Writing – review & editing. **Mostafa Pirmoradian:** Writing – review & editing, Conceptualization, Data curation, Formal analysis, Supervision, Investigation, Writing – original draft, Writing – review & editing.

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.

Data availability

No data was used for the research described in the article.

References

[1] S.U. Choi, J.A. Eastman, Enhancing Thermal Conductivity of Fluids with Nanoparticles, Argonne National Lab.(ANL), Argonne, IL (United States), 1995.

[2] S.Z. Heris, H. Bagheri, S.B. Mousavi, S. Hosseini Nami, Optimizing nanofluid additives for enhanced thermophysical properties in anionic crude oil for EOR applications, Can. J. Chem. Eng. 102 (7) (2024) 2418–2431.

[3] Z. Karimi Shoar, H. Pourpasha, S. Zeinali Heris, S.B. Mousavi, M. Mohammadpourfard, The effect of heat transfer characteristics of macromolecule fouling on heat exchanger surface: a dynamic simulation study, Can. J. Chem. Eng. 101 (10) (2023) 5802–5817.

[4] H. Pourpasha, S.Z. Heris, S.B. Mousavi, Thermal performance of novel ZnFe2O4 and TiO2-doped MWCNT nanocomposites in transformer oil, J. Mol. Liq. 394 (2024) 123727.

[5] K. Bashirnezhad, et al., Viscosity of nanofluids: a review of recent experimental studies, Int. Commun. Heat Mass Tran. 73 (2016) 114–123.

[6] H.D. Koca, et al., Effect of particle size on the viscosity of nanofluids: a review, Renew. Sustain. Energy Rev. 82 (2018) 1664–1674.

[7] S.S. Murshed, P. Estellé, A state of the art review on viscosity of nanofluids, Renew. Sustain. Energy Rev. 76 (2017) 1134–1152.

[8] K. Selvarajoo, V.V. Wanatasanappan, N.Y. Luon, Experimental measurement of thermal conductivity and viscosity of Al2O3-GO (80: 20) hybrid and mono nanofluids: a new correlation, Diam. Relat. Mater. 144 (2024) 111018.

[9] O. Khouri, H.R. Goshayeshi, S.B. Mousavi, S. Hosseini Nami, S. Zeinali Heris, Heat transfer enhancement in industrial heat exchangers using graphene oxide nanofluids, ACS Omega 9 (22) (2024) 24025–24038.

[10] A.M. Ajeena, I. Farkas, P. Víg, Characterization, rheological behaviour, and dynamic viscosity of ZrO2-SiC (50–50)/DW hybrid nanofluid under different temperatures and solid volume fractions: an experimental study and proposing a new correlation, Powder Technol. 431 (2024) 119069.

[11] M. Sepehrnia, et al., Experimental study, prediction modeling, sensitivity analysis, and optimization of rheological behavior and dynamic viscosity of 5W30 engine oil based SiO2/MWCNT hybrid nanofluid, Ain Shams Eng. J. 15 (1) (2024) 102257.

[12] V.V. Wanatasanappan, et al., Viscosity and rheological behavior of Al2O3-Fe2O3/water-EG based hybrid nanofluid: a new correlation based on mixture ratio, J. Mol. Liq. 375 (2023) 121365.

[13] M. Sepehrnia, et al., Experimental study on the dynamic viscosity of hydraulic oil HLP 68-Fe3O4-TiO2-GO ternary hybrid nanofluid and modeling utilizing machine learning technique, J. Taiwan Inst. Chem. Eng. 145 (2023) 104841.

[14] M.H. Esfe, D. Toghraie, E.M. Ardeshiri, Experimental study of rheological behavior of MWCNT (50%)-MgO (50%)/SAE40 hybrid nanofluid: dynamic viscosity optimization and numerical simulation of turbulent flow, Ann. Nucl. Energy 182 (2023) 109575.

[15] A.M. Ajeena, I. Farkas, P. Víg, A comparative experimental investigation of dynamic viscosity of ZrO2/DW and SiC/DW nanofluids: characterization, rheological behavior, and development of new correlation, Heliyon 9 (10) (2023) e21113.

[16] M.H. Esfe, et al., Presenting a model based on knowledge management in modeling and optimizing the dynamic viscosity of MWCNT (30%)-ZnO (70%)/SAE 40 oil hybrid nano-lubricant with response surface methodology (RSM) for industrial uses, Tribol. Int. 191 (2024) 109135.

[17] A.R. Abbasian, et al., Experimental study of preparing the CoFe2O4 magnetic nanofluid and measuring thermal-fluid characteristics of the stabilized magnetocaloric nanofluid, Mater. Sci. Eng., B 306 (2024) 117462.

[18] A. Hafeez, et al., Heat transfer performance in a Hybrid nanofluid (Cu-Al2O3/kerosene oil) flow over a shrinking cylinder, Case Stud. Therm. Eng. 52 (2023) 103539.

[19] M.H. Esfe, et al., Using the RSM to evaluate the rheological behavior of SiO2 (60%)-MWCNT (40%)/SAE40 oil hybrid nanofluid and investigating the effect of different parameters on the viscosity, Tribol. Int. 184 (2023) 108479.

[20] G.A. Bekey, K.Y. Goldberg, Neural Networks in Robotics, vol. 202, Springer Science & Business Media, 2012.

[21] M. Zeinaddini, M. Pirmoradian, F. Azimifar, Optimizing the torque of knee movements of a rehabilitation robot, J. Simul. Anal. Novel Technol. Mech. Eng. 11 (1) (2018) 5–14.

[22] A. Alfaleh, et al., Predicting thermal conductivity and dynamic viscosity of nanofluid by employment of Support Vector Machines: a review, Energy Rep. 10 (2023) 1259–1267.

[23] M.H. Esfe, F. Amoozadkhalili, D. Toghraie, Determining the optimal structure for accurate estimation of the dynamic viscosity of oil-based hybrid nanofluid containing MgO and MWCNTs nanoparticles using multilayer perceptron neural networks with Levenberg-Marquardt Algorithm, Powder Technol. 415 (2023) 118085.

[24] Z. Wang, et al., Using different Heuristic strategies and an adaptive Neuro-Fuzzy inference system for multi-objective optimization of Hybrid Nanofluid to provide an efficient thermal behavior, Swarm Evol. Comput. 86 (2024) 101536.

[25] T. Zhang, et al., Optimization of thermophysical properties of nanofluids using a hybrid procedure based on machine learning, multi-objective optimization, and multi-criteria decision-making, Chem. Eng. J. 485 (2024) 150059.

[26] J. Gao, et al., An RBF-based artificial neural network for prediction of dynamic viscosity of MgO/SAE 5W30 oil hybrid nano-lubricant to obtain the best performance of energy systems, Mater. Today Commun. 38 (2024) 107836.

[27] B.M. Fadhl, et al., Dynamic viscosity modeling of nanofluids with MgO nanoparticles by utilizing intelligent methods, Energy Rep. 9 (2023) 5397–5403.

[28] Y. AbuShanab, et al., Accurate prediction of dynamic viscosity of polyalpha-olefin boron nitride nanofluids using machine learning, Heliyon 9 (6) (2023) e16716.

[29] S. Hua, et al., Dynamic viscosity prediction using artificial intelligence for an antifreeze containing MWCNT–alumina hybrid nanopowders, Eng. Appl. Artif. Intell. 126 (2023) 107046.

[30] M.H. Esfe, et al., 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.

[31] 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. Appl. Artif. Intell. 122 (2023) 106107.

[32] S. Singh, S.K. Ghosh, A unique artificial intelligence approach and mathematical model to accurately evaluate viscosity and density of several nanofluids from experimental data, Colloids Surf. A Physicochem. Eng. Asp. 640 (2022) 128389.

[33] Z. Said, et al., Synthesis, stability, density, viscosity of ethylene glycol-based ternary hybrid nanofluids: experimental investigations and model-prediction using modern machine learning techniques, Powder Technol. 400 (2022) 117190.

[34] P.K. Kanti, et al., Explainable machine learning techniques for hybrid nanofluids transport characteristics: an evaluation of shapley additive and local interpretable model-agnostic explanations, J. Therm. Anal. Calorim. 149 (21) (2024) 11599–11618.

[35] P.K. Kanti, et al., Experimental and explainable machine learning approach on thermal conductivity and viscosity of water based graphene oxide based mono and hybrid nanofluids, Sci. Rep. 14 (1) (2024) 30967.

[36] S. Akilu, et al., Machine learning analysis of thermophysical and thermohydraulic properties in ethylene glycol-and glycerol-based SiO2 nanofluids, Sci. Rep. 14 (1) (2024) 14829.

[37] K. Sharma, et al., Prognostic modeling of polydisperse SiO2/Aqueous glycerol nanofluids' thermophysical profile using an explainable artificial intelligence (XAI) approach, Eng. Appl. Artif. Intell. 126 (2023) 106967.

[38] R. Rostamzadeh-Renani, et al., Multi-objective optimization of rheological behavior of nanofluids containing CuO nanoparticles by NSGA II, MOPSO, and MOGWO evolutionary algorithms and Group Method of Data Handling Artificial neural networks, Mater. Today Commun. 38 (2024) 107709.

[39] R. Rostamzadeh-Renani, et al., Prediction of the thermal behavior of multi-walled carbon nanotubes-CuO-CeO2 (20-40-40)/water hybrid nanofluid using different types of regressors and evolutionary algorithms for designing the best artificial neural network modeling, Alex. Eng. J. 84 (2023) 184–203.
[40] M. Baghoolizadeh, et al., Prediction and extensive analysis of MWCNT-MgO/oil SAE 50 hybrid nano-lubricant rheology utilizing machine learning and genetic algorithms to find ideal attributes, Tribol. Int. (2024) 109582.
[41] M. Baghoolizadeh, et al., Using different machine learning algorithms to predict the rheological behavior of oil SAE40-based nano-lubricant in the presence of MWCNT and MgO nanoparticles, Tribol. Int. 187 (2023) 108759.
[42] M. Baghoolizadeh, et al., Using of artificial neural networks and different evolutionary algorithms to predict the viscosity and thermal conductivity of silica-alumina-MWCN/water nanofluid, Heliyon 10 (4) (2024) e26279.
[43] W. Jin, et al., Regression modeling and multi-objective optimization of rheological behavior of non-Newtonian hybrid antifreeze: using different neural networks and evolutionary algorithms, Int. Commun. Heat Mass Tran. 155 (2024) 107578.
[44] M. Sabah, et al., 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. Petrol. Sci. Eng. 177 (2019) 236–249.
[45] Y. Chen, et al., Evaluation efficiency of hybrid deep learning algorithms with neural network decision tree and boosting methods for predicting groundwater potential, Geocarto Int. 37 (19) (2022) 5564–5584.
[46] J.S. Raj, J.V. Ananthi, Recurrent neural networks and nonlinear prediction in support vector machines, J. Soft Comput. Paradigm (JSCP) 1 (1) (2019) 33–40.
[47] 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.
[48] I. Goodfellow, Deep Learning, MIT press, 2016.
[49] C.M. Bishop, N.M. Nasrabadi, Pattern Recognition and Machine Learning, vol. 4, Springer, 2006.
[50] S. Haykin, Neural Networks: a Comprehensive Foundation, Prentice Hall PTR, 1998.
[51] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Learning representations by back-propagating errors, Nature 323 (6088) (1986) 533–536.
[52] 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.
[53] 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 (4) (2019) 1–12.
[54] M. Gumus, M.S. Kiran, Crude oil price forecasting using XGBoost, in: 2017 International Conference on Computer Science and Engineering (UBMK), IEEE, 2017.
[55] J. Chen, et al., Improved XGBoost model based on genetic algorithm, Int. J. Comput. Appl. Technol. 62 (3) (2020) 240–245.
[56] A. El-Saleh, et al., Mean opinion score estimation for mobile broadband networks using bayesian networks, CMC-COMPUTERS MATERIALS & CONTINUA 72 (3) (2022).
[57] B.G. Marcot, T.D. Penman, Advances in Bayesian network modelling: integration of modelling technologies, Environ. Model. Software 111 (2019) 386–393.
[58] B. Cai, et al., Application of Bayesian networks in reliability evaluation, IEEE Trans. Ind. Inf. 15 (4) (2018) 2146–2157.
[59] J.-S.R. Jang, Fuzzy modeling using generalized neural networks and kalman filter algorithm, in: AAAI, 1991.
[60] J.-S. Jang, ANFIS: adaptive-network-based fuzzy inference system, IEEE transactions on systems, man, and cybernetics 23 (3) (1993) 665–685.
[61] A. Abraham, Adaptation of fuzzy inference system using neural learning, in: Fuzzy Systems Engineering, Springer, 2005, pp. 53–83.
[62] D. Karaboga, E. Kaya, Adaptive network based fuzzy inference system (ANFIS) training approaches: a comprehensive survey, Artif. Intell. Rev. 52 (4) (2019) 2263–2293.
[63] C. Nwankpa, C. Nwankpa, et al., Activation functions: comparison of trends in practice and research for deep learning, arXiv preprint (2018) arXiv:1811.03378, https://doi.org/10.48550/arXiv.1811.03378.
[64] H. Zhou, et al., Combination of group method of data handling neural network with multi-objective gray wolf optimizer to predict the viscosity of MWCNT-TiO2-oil SAE50 nanofluid, Case Stud. Therm. Eng. 64 (2024) 105541.
[65] S.A. Hussein, et al., Applying different machine learning algorithms to predict the viscosity behavior of MWCNT–alumina/water–ethylene glycol (80: 20) hybrid antifreeze, Int. J. Thermofluids 24 (2024) 100966.
[66] K.E. Taylor, Summarizing multiple aspects of model performance in a single diagram, J. Geophys. Res. Atmos. 106 (D7) (2001) 7183–7192.
[67] 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.
[68] M. Bahrami, M. Akbari, A. Karimipour, M. Afrand, An experimental study on rheological behavior of hybrid nanofluids made of iron and copper oxide in a binary mixture of water and ethylene glycol: non-Newtonian behavior, Exp. Therm. Fluid Sci. 79 (2016) 231–237.