Bulletin of the National Research Centre

RESEARCH
Open Access

Modeling nanofluid viscosity: comparing models and optimizing feature selection—a novel approach

Ekene Onyiriuka[1*]

Abstract

Background The accurate prediction of viscosity in nanofluids is essential for comprehending their flow behavior and enhancing their effectiveness in different industries. This research delves into modeling the viscosity of nanofluids and assessing various models through cross-validation techniques. The models are compared based on the root mean square error of the cross-validation sets, which served as the selection criteria.

The main body of the abstract Four feature selection algorithms namely the minimum redundancy maximum relevance, F-test, RReliefF were evaluated to identify the most influential features for viscosity prediction. The feature selection based on physical meaning was the algorithm that yielded the best results, as outlined in this study. This methodology takes into account the physical relevance of most aspects of the nanofluid's viscosity. To assess the predictive performance of the models, a cross-validation process was conducted, which provided a robust evaluation. The root mean squared error of the validation sets was used to compare the models. This rigorous evaluation identified the most accurate and reliable model for predicting nanofluid viscosity.

Results The results showed that the novel feature selection algorithm outclassed the established approaches in predicting the viscosity of single material nanofluid. The proposed feature selection algorithm had a root mean squared error of 0.022 and an r squared value of 0.9941 for the validation set, while for the test set, the root mean squared error was 0.0146, the mean squared error was 0.0157, the r squared value was 0.9924.

Conclusions This research provides valuable insights into nanofluid viscosity and offers guidance on choosing the most suitable features for viscosity modeling. The study also highlights the importance of using physical meaning to select features and cross-validation to assess model performance. The models developed in this study can be helpful in predicting nanofluid viscosity and optimizing their use in different industrial processes.

Keywords Nanofluids, Viscosity prediction, Modeling, Feature selection, Cross-validation, Root mean square error

Background

Predicting viscosity in nanofluids plays a crucial role in understanding their flow behavior and optimizing their applications in various industries (Bhaumik et al. 2023; Chiniforooshan Esfahani 2023; Esfe and Arani 2018; Gholizadeh et al. 2020; Onyiriuka 2023b; Said et al. 2021; Tan et al. 2022; Yadav et al. 2020). Nanofluids, suspensions of nanoparticles in base fluids, exhibit unique rheological properties that differ from those of conventional fluids (Tan et al. 2022). Accurate prediction of the viscosity of nanofluids is essential for the efficient design and optimization of heat transfer systems, lubrication processes, and other applications.

As a critical step in the modeling process, feature selection aims to identify the most influential features

*Correspondence:
Ekene Onyiriuka
mnejo@leeds.ac.uk
[1] School of Mechanical Engineering, University of Leeds, Leeds LS2 9JT, UK

contributing to nanofluids' viscosity. It involves selecting relevant input variables or features from potential predictors. This study focuses on the feature selection process for predicting the viscosity of single material nanofluids. Single material nanofluids consist of nanoparticles and base fluid that are stably mixed. A nanofluid viscosity model provides a unique system for investigating the impact of various parameters on viscosity. By carefully selecting the appropriate features, we can uncover the underlying relationships between the composition, particle size, temperature, other factors, and the resulting viscosity of nanofluids.

The objective of this study is to investigate various feature selection methods and pinpoint the primary factors that have a significant impact on the viscosity of single material nanofluids. By utilizing physical, sophisticated statistical, and machine learning techniques, the goal is to create precise prediction models that can estimate the viscosity of nanofluids based on a chosen set of input features.

The findings of this study will contribute to a deeper understanding of the factors that govern the viscosity of nanofluids and provide valuable insights for optimizing their performance in practical applications. Moreover, the developed feature selection techniques can be applied to other nanofluid systems, enabling efficient and effective viscosity prediction models for various nanofluid applications.

Various researchers have studied this subject extensively but mainly focusing on its accuracy than its generality and conventional feature selection. Gholizadeh et al. (2020) in 2020, a group of researchers—Gholizadeh, Jamei, Ahmadianfar, and Pourrajab—conducted a study on predicting the viscosity of nanofluids using the Random Forest (RF) approach. What was unique about their research is that they utilized the RF method to estimate the thermophysical property of nanofluids for the very first time. The study focused on five significant parameters, which included volume fraction, nanoparticle size, nanoparticle density, and base fluid viscosity.

The researchers used various statistical tools to compare different correlations and found that their model was the best, with an R^2 of 0.9972. The next best was Nguyen's model with an R^2 of 0.654, followed by the Maiga et al. correlation at an R^2 of 0.652 (Gholizadeh et al. 2020).

It's worth noting that there was no validation data set mentioned for their case. The researchers also utilized the out-of-bag error rate method to tune the number of trees and predictors of the RF model. Lastly, they applied a performance index to compare different machine learning models accurately. However, the paper did not consider the application of cross-validation in comparing models, Brownlee (2016), states that from a machine learning viewpoint, it is an essential step in model evaluation and comparison.

It was observed from the study that the volume fraction increased viscosity while particle size decreased it. The nanoparticle volume fraction was noticed to have the most significant impact in predicting the viscosity of nanofluids, while the temperature had the least predictive impact (Gholizadeh et al. 2020).

Rudyak and Minakov (2018) stated that a universal formula describing the viscosity coefficient of any nanofluid has yet to be derived. In addition, most measurements of this quantity have mainly led to opposite results. Einstein and other researchers, including the international nanofluid properties benchmark exercise (Buongiorno et al. 2009; Kim et al. 2009; Venerus et al. 2010), thought that the volume fraction was the sole determining factor of nanofluids' viscosity. It has now been shown that the non-universality models are because the volume fraction of the nanoparticles is not the only factor determining nanofluids' viscosity.

According to a recent study, the size and material of nanoparticles play a significant role in determining the viscosity of nanofluids. As the concentration of particles increases, the viscosity of nanofluids also increases, while an increase in particle size or temperature results in a decrease in viscosity. Additionally, the type of nanoparticle used can lead to a significant difference in viscosity. Nanofluids have been found to have higher viscosity levels than ordinary fluids with coarse dispersion (Rudyak and Minakov 2018).

The viscosity of nanofluids can be estimated using the modified Einstein's quadratic model form for low and moderate concentrations of nanoparticles. However, the coefficients in this equation vary based on the material and size of the particles. Increasing the degree of order in a fluid lead to an increase in effective viscosity, which can be achieved by decreasing the particle size and increasing the particle concentration (Rudyak and Minakov 2018).

Nanofluids are more ordered than base fluids, and the addition of nanoparticles helps to improve momentum transfer. Molecular dynamics suggest that nanoparticle–molecule interaction is the primary reason for increased viscosity in nanofluids. Einstein's equations do not apply to nanofluids due to assumptions like neglecting interactions between molecules and nanoparticles, creeping flows, or very low particle Reynolds numbers. Therefore, further investigation is needed to understand the relationship between the viscosity of nanofluids and nanoparticle materials, as concluded by the study (Rudyak and Minakov 2018).

Machine learning models

In this study several machine learning models were applied, namely: The Gaussian process regressor, Neural network, support vector machines, decision trees, ensembles, and linear regression.

The Gaussian process regressor uses probability distributions to model relationships between variables. The neural network learns complex patterns of data through layers of interconnected nodes. Support vector machines finds a hyperplane that separates data into classes. Decision trees divides data into subsets based on feature threshold. Ensemble models combine multiple models to improve predictive accuracy and robustness. The linear regression establishes a linear relationship between features and target (Mahesh 2020; Sarker 2021).

Data collection and analysis

The data were collected from open literature: ZnO—ethylene glycol (Lee et al. 2012), TiO_2—water, $Mg(OH)_2$—ethylene glycol (Esfandiary et al. 2016), Al_2O_3—water (Nguyen et al. 2008), SiO_2—water (Tavman et al. 2008), CuO—water (Pastoriza-Gallego et al. 2011), CuO—ethylene glycol (Yadav et al. 2020), Al_2O_3–water (Pastoriza-Gallego et al. 2009), Al_2O_3—ethylene glycol (Yadav et al. 2020), CeO_2—ethylene glycol (Yadav et al. 2020). The total number of data rows collected was 245, with 20 columns including the response variable. There were no missing data points in the data set; hence, the study did not need to impute missing values or drop incomplete rows of data.

The variables are represented by the following nomenclature for ease of reference, as shown in Table 1.

Table 1 Variables nomenclature for ease of reference (Onyiriuka 2023a)

Abbreviations	Full names
TC	Nanofluid temperature (°C)
DP	Particle size diameter (nm)
VF	Volume fraction (%)
NPk	Nanoparticle thermal conductivity (W/(m K))
NPd	Nanoparticle density (kg/m^3)
NPa	Nanoparticle thermal diffusivity (m^2/s) e+07
NPcp	Nanoparticle-specific heat capacity (J/(kg K))
NPmp	Nanoparticle melting point (°C)
NPde	Nanoparticle dielectric constant (–)
NPri	Nanoparticle refractive index (–)
BFk	Base fluid thermal conductivity (W/(m K))
BFd	Base fluid density (kg/m^3)
BFa	Base fluid thermal diffusivity (m^2/s) e+07
BFv	Base fluid viscosity (Pa·s)
BFkv	Base fluid kinematic viscosity (m^2/s) e+07
BFcp	Base fluid specific heat capacity (J/(kg K))
BFbp	Base fluid boiling point (°C)
BFst	Base fluid surface tension (mN/m)
BFde	Base fluid dielectric constant (–)
NFv	Nanofluid viscosity (Pa·s)

In the provided Fig. 1a, we can observe the distribution of each variable. However, there seems to be no normality in general for any of the variables. Each plot in the figure represents a histogram plot displaying the range of values for each feature.

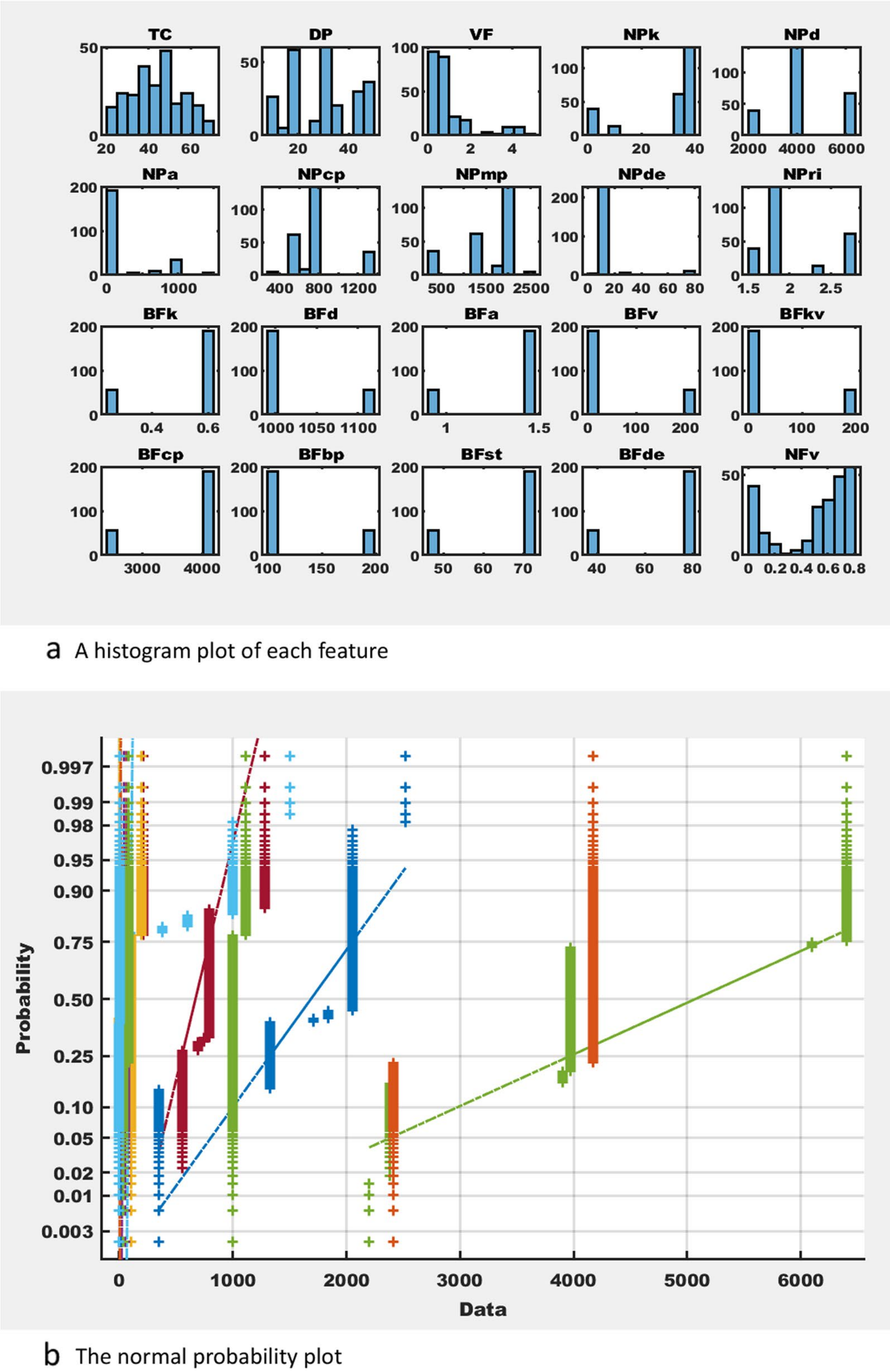
For instance, the temperature values are plotted on the x-axis, while the frequency of each temperature value is represented on the y-axis. The first plot in Fig. 1a shows the temperature values, where the most frequently occurring temperature value is 50 °C. On the other hand, the least occurring temperature value of 70 °C was also the highest temperature value. The temperature values between 35 and 45 °C were the most frequently occurring groups in the data set. The general trend in the data shows a rise in the beginning and a fall toward the end. Similar analysis can be seen for the other features. This property is also illustrated clearly in the standard probability plot in Fig. 1b.

In Fig. 1b, we can see a normal probability plot that compares the distribution of data in each variable to the standard normal distribution. The plot uses plus sign markers ('+') to represent each data point in each variable. Two reference lines are drawn to show the theoretical normal distribution. The first reference line is a solid line that connects the data's first and third quartiles, while the second is a dashed line that extends the solid line to the ends of the data range. If the data follows a normal distribution, the points align along the reference line.

However, if the data deviate from the normal distribution, it introduces a curvature or deviation in the plot, indicating that the data distribution differs from the expected normal distribution (MathWorks 2022). By visually inspecting the standard probability plot in Fig. 1b, we can observe the departure from normality and the nature of the data distribution.

Figure 2 shows the box plot of each variable.

Using a five-number summary, box plots are a common method for displaying data distribution. The temperature data's box plot shows the minimum, first quartile, median, third quartile, and maximum values of temperature. The five components make up the box plots, providing information about the temperature distribution for instance. These components include the median, hinges (Q1 and Q3 quartiles), fences (adjacent extremes), whiskers (minimum and maximum values, excluding outliers), and outliers (data points outside the whiskers).



a A histogram plot of each feature

b The normal probability plot

Fig. 1 **a** A histogram plot of each feature. **b** The normal probability plot

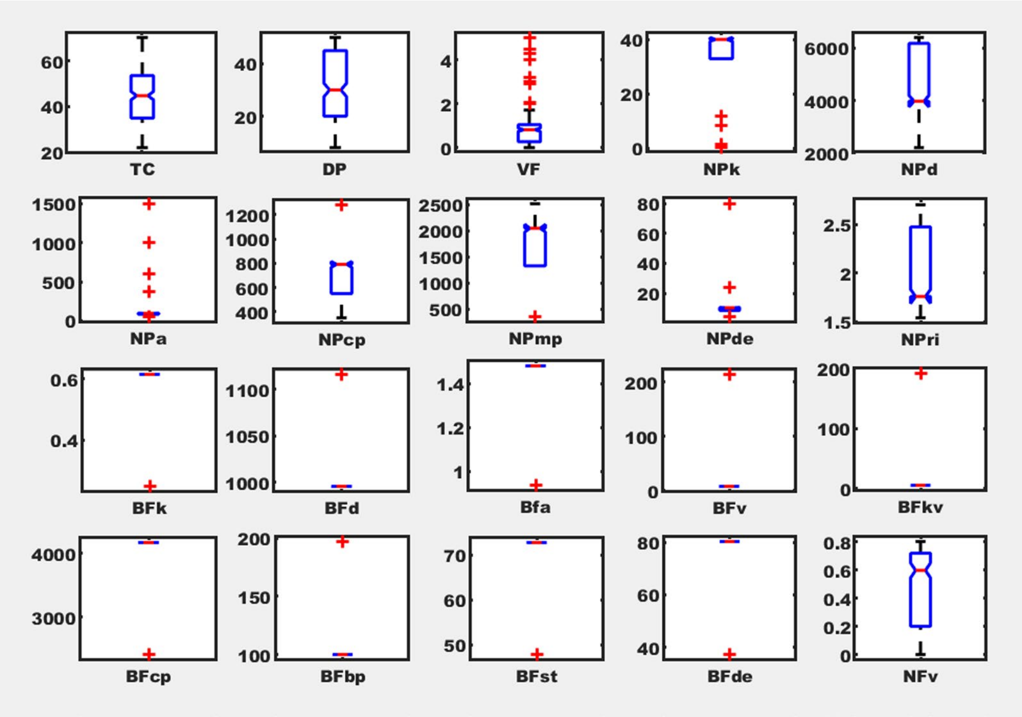


Fig. 2 A box plot of each feature

Notched box plots, narrow the box around the median to provide an approximate 95% confidence interval for the population's median. Notches are particularly useful for evaluating the significance of differences between medians. In Fig. 2, it was observed that notches of the temperature values and the particle size overlap signifying the similar median distribution. The height of the notches is proportional to the interquartile range (IQR) of the sample and inversely proportional to the square root of the sample size. By analyzing the plot, it is evident that each variable has distinct values except for the thermal conductivity, thermal diffusivity, specific heat capacity, surface tension, and dielectric constant of the base fluid, which are similar but opposite to the density, viscosity, kinematic viscosity, and boiling point of the base fluid. To model the viscosity of nanofluids, it is recommended to explore decision trees, ensemble models, and neural networks.

Methods

This section tests various modeling and feature selection algorithms, including the algorithm outlined below in Sect. "Algorithm for parameter selection applied for viscosity" [Novel Feature selection algorithms (NFSA)]. The other investigated feature selection algorithms include minimum redundancy, maximum relevance (MRMR), FTest, and RReliefF. Tables 2 and 3 summarize the results obtained by applying these algorithms.

The Minimum Redundancy Maximum Relevance (MRMR)

The MRMR algorithm is a technique used in machine learning and data mining to select a subset of features from a larger set. The main objective of this algorithm is to maximize the relevance of the chosen features to the target variable while minimizing redundancy among them. Here is how the MRMR algorithm works (ÇALIŞKAN 2023; Sakthivel et al. 2023; TM & VENI 2023):

First, start with an empty set of selected features. Then, calculate the relevance of each feature by using different metrics such as mutual information, correlation coefficient, or information gain, with respect to the target variable. Next, select the feature with the highest relevance and add it to the selected feature set. After that, for every remaining feature, calculate its redundancy with respect to the already selected features. Redundancy is a measure of how much information a feature provides beyond what is already captured by the selected features (TM & VENI 2023).

Calculate the MRMR score for each feature by subtracting its redundancy from its relevance. Choose the feature with the highest MRMR score and add it to the selected feature set. Repeat the steps until the desired

Table 2 Models performance and comparison

Model type	Preset	RMSE (validation)	MSE (validation)	R^2 (validation)	MAE (validation)	MAE (test)	MSE (test)	RMSE (Test)	R^2 (test)
Gaussian process regression	Custom Gaussian process regression	0.022	0.0005	0.9941	0.0146	0.0157	0.0004	0.0211	0.9924
Gaussian process regression	Custom Gaussian process regression	0.0228	0.0005	0.9936	0.0129	0.0186	0.0006	0.0251	0.9894
Gaussian process regression	Custom Gaussian process regression	0.0232	0.0005	0.9934	0.0131	0.0186	0.0006	0.0251	0.9894
Gaussian process regression	Custom Gaussian process regression	0.0263	0.0007	0.9915	0.0144	0.0237	0.0012	0.034	0.9805
Gaussian process regression	Exponential GPR	0.0302	0.0009	0.9889	0.0172	0.0201	0.0008	0.0278	0.987
Gaussian process regression	Rational quadratic GPR	0.0341	0.0012	0.9858	0.0192	0.0203	0.0008	0.0283	0.9864
Gaussian process regression	Matern 5/2 GPR	0.0359	0.0013	0.9842	0.0208	0.0229	0.0009	0.0298	0.985
Neural network	Medium neural network	0.0363	0.0013	0.9839	0.0222	0.0372	0.0034	0.0585	0.9422
Neural network	Bilayered neural network	0.0367	0.0013	0.9835	0.0218	0.0176	0.0005	0.0226	0.9914
Neural network	Trilayered neural network	0.0388	0.0015	0.9816	0.023	0.0449	0.0051	0.0713	0.914
Gaussian process regression	Squared exponential GPR	0.0431	0.0019	0.9773	0.0256	0.0277	0.0015	0.0393	0.9738
Neural network	Wide neural network	0.0438	0.0019	0.9765	0.025	0.0374	0.0027	0.0521	0.9541
Neural network	Narrow neural network	0.052	0.0027	0.9669	0.0293	0.0408	0.0039	0.0621	0.9348
Linear regression	Interactions Linear	0.0535	0.0029	0.965	0.0371	0.0396	0.0025	0.0495	0.9585
Stepwise linear regression	Stepwise linear	0.0554	0.0031	0.9624	0.0394	0.0388	0.0028	0.0528	0.9529
SVM	Medium Gaussian SVM	0.0582	0.0034	0.9585	0.0438	0.0451	0.0026	0.0514	0.9552
Tree	Fine tree	0.0592	0.0035	0.9571	0.0368	0.0328	0.0022	0.0464	0.9636
SVM	Quadratic SVM	0.0617	0.0038	0.9534	0.0476	0.0349	0.0021	0.046	0.9642
Linear regression	Linear	0.0642	0.0041	0.9496	0.0504	0.0416	0.0028	0.0529	0.9526
Linear regression	Robust linear	0.0653	0.0043	0.9478	0.0511	0.0406	0.0026	0.0513	0.9554
SVM	Linear SVM	0.0658	0.0043	0.947	0.0517	0.0411	0.0026	0.0511	0.9558
SVM	Fine Gaussian SVM	0.0717	0.0051	0.9371	0.0518	0.0494	0.0037	0.0611	0.9368
SVM	Cubic SVM	0.075	0.0056	0.9311	0.0545	0.0393	0.0038	0.0617	0.9356
Gaussian process regression	Custom Gaussian process regression	0.0819	0.0067	0.9179	0.0389	0.0549	0.0121	0.1099	0.7955
Ensemble	Boosted trees	0.0819	0.0067	0.9178	0.0495	0.0315	0.0019	0.0435	0.968
Gaussian process regression	Custom Gaussian process regression	0.0877	0.0077	0.9058	0.0663	0.0801	0.0093	0.0965	0.8425
SVM	Coarse Gaussian SVM	0.0945	0.0089	0.8906	0.0702	0.0604	0.0046	0.0677	0.9225
Ensemble	Bagged trees	0.1021	0.0104	0.8724	0.0641	0.0593	0.0091	0.0954	0.8459

Table 2 (continued)

Model type	Preset	RMSE (validation)	MSE (validation)	R^2 (validation)	MAE (validation)	MAE (test)	MSE (test)	RMSE (Test)	R^2 (test)
Kernel	SVM kernel	0.1349	0.0182	0.7772	0.0947	0.1012	0.0235	0.1534	0.602
Kernel	Least squares regression kernel	0.1473	0.0217	0.7343	0.117	0.1303	0.0247	0.1572	0.5817
Tree	Medium tree	0.1504	0.0226	0.7232	0.0941	0.0546	0.0165	0.1286	0.7204
Tree	Coarse tree	0.2349	0.0552	0.3243	0.1742	0.0967	0.0137	0.1171	0.7679

number of features is selected or a stopping criterion is met (for example a predefined threshold for MRMR score). The final selected features are those in the selected feature set. The MRMR algorithm aims to balance between informative features (high relevance) and avoiding redundant information. By using this approach, the algorithm can help improve the efficiency and interpretability of machine learning models by reducing the dimensionality of the input feature space while retaining the most relevant information (TM & VENI 2023).

FTest

The F-test algorithm is a statistical technique that identifies the features with the most relevance or discriminatory power for a given target variable (Mathew 2023; Venkatesan 2023). For each feature in the dataset, the F-statistic is calculated to determine the ratio of between-class variability to within-class variability. The corresponding p value is computed to represent the likelihood of obtaining the observed F-statistic by chance. The features are then sorted based on their F-statistic or p value in ascending or descending order. The top-k features with the highest F-statistic or lowest p value are selected as the final feature subset (Mathew 2023; Venkatesan 2023).

By examining the variability between different classes and within each class, the F-test algorithm assesses the relationship between each feature and the target variable. Features with higher F-statistics or lower p values indicate stronger associations with the target variable. The F-test algorithm aids in identifying the most relevant features for a given classification or regression task by selecting the features with the highest discriminatory power (Mathew 2023; Venkatesan 2023).

RReliefF

The RReliefF algorithm is a technique for selecting features that can effectively differentiate between instances of different classes (Aggarwal et al. 2023). It assigns weights to each feature based on its discriminatory power. The weights are updated iteratively and aggregated across all instances to identify the most relevant features for classification tasks. The selected features are those with the highest scores, indicating their importance in separating instances of different classes (Aggarwal et al. 2023).

To begin, the weights for each instance are initialized to zero. For each instance in the dataset, the weight updates are calculated by considering the differences between the feature values of the current instance and its closest instances of the same and different classes. The weights are then updated accordingly, with greater emphasis placed on features that contribute more to distinguishing between instances of different classes. The feature scores are calculated by aggregating the weight updates across all instances. Finally, the top-k features with the highest scores are selected as the final feature subset (Aggarwal et al. 2023).

Algorithm for parameter selection applied for viscosity

Here we discuss the procedure for selecting parameters according to the novel method discussed by (Onyiriuka 2023a) for predicting the viscosity of single material nanofluids.

(1) Check the problem being solved.
(2) List all the possible features.
(3) Drop features that have no meaning or direct implication to the viscosity of a fluid. For example, using single material nanofluids:

  (a) Fluid features—Temperature
  (b) Multiphase features—Volume fraction and particle size
  (c) Material features

    (i) Nanoparticle material: Any two intensive properties will fix the material of the nanoparticle type (Callister 2007; Cengel et al. 2011; Moran et al. 2010).
(ii) Base fluid material: Any two intensive properties will fix the material of the base fluid type (Callister 2007; Cengel et al. 2011; Moran et al. 2010).

So, these three feature groupings define a nanofluid.

Table 3 Model parameters and the optimized Gaussian process model

Preset	Hyperparameters	Selected features	Feature ranking algorithm	Optimizer options
Custom Gaussian process regression	Signal standard deviation: 0.20121; Optimize numeric parameters: Yes	07/19 TC, DP, VF, NPcp, NPde, BFd, BFcp	Novel Feature selection algorithms (NFSA)	Optimizer: Bayesian optimization; Acquisition function: Expected improvement per second plus; Iterations: 30; Training time limit: false
Custom Gaussian process regression	Signal standard deviation: 0.20121; Optimize numeric parameters: Yes	19/19	None	Optimizer: Bayesian optimization; Acquisition function: Expected improvement per second plus; Iterations: 30; Training time limit: false
Custom Gaussian process regression	Signal standard deviation: 0.20121; Optimize numeric parameters: Yes	19/19	FTest	Optimizer: Bayesian optimization; Acquisition function: Expected improvement per second plus; Iterations: 30; Training time limit: false
Custom Gaussian process regression	Signal standard deviation: 0.20121; Optimize numeric parameters: Yes	19/19	MRMR	Optimizer: Bayesian optimization; Acquisition function: Expected improvement per second plus; Iterations: 30; Training time limit: false
Exponential GPR	Basis function: Constant; Kernel function: Exponential; Use isotropic kernel: Yes; Kernel scale: Automatic; Signal standard deviation: Automatic; Sigma: Automatic; Standardize data: Yes; Optimize numeric parameters: Yes	19/19	None	Not applicable
Rational quadratic GPR	Basis function: Constant; Kernel function: Rational Quadratic; Use isotropic kernel: Yes; Kernel scale: Automatic; Signal standard deviation: Automatic; Sigma: Automatic; Standardize data: Yes; Optimize numeric parameters: Yes	19/19	None	Not applicable
Matern 5/2 GPR	Basis function: Constant; Kernel function: Matern 5/2; Use isotropic kernel: Yes; Kernel scale: Automatic; Signal standard deviation: Automatic; Sigma: Automatic; Standardize data: Yes; Optimize numeric parameters: Yes	19/19	None	Not applicable
Medium neural network	Number of fully connected layers: 1; First layer size: 25; Activation: ReLU; Iteration limit: 1000; Regularization strength (Lambda): 0; Standardize data: Yes	19/19	None	Not applicable
Bilayered neural network	Number of fully connected layers: 2; First layer size: 10; Second layer size: 10; Activation: ReLU; Iteration limit: 1000; Regularization strength (Lambda): 0; Standardize data: Yes	19/19	None	Not applicable

Table 3 (continued)

Preset	Hyperparameters	Selected features	Feature ranking algorithm	Optimizer options
Trilayered Neural Network	Number of fully connected layers: 3; First layer size: 10; Second layer size: 10; Third layer size: 10; Activation: ReLU; Iteration limit: 1000; Regularization strength (Lambda): 0; Standardize data: Yes	19/19	None	Not applicable
Squared exponential GPR	Basis function: Constant; Kernel function: Squared Exponential; Use isotropic kernel: Yes; Kernel scale: Automatic; Signal standard deviation: Automatic; Sigma: Automatic; Standardize data: Yes; Optimize numeric parameters: Yes	19/19	None	Not applicable
Wide neural network	Number of fully connected layers: 1; First layer size: 100; Activation: ReLU; Iteration limit: 1000; Regularization strength (Lambda): 0; Standardize data: Yes	19/19	None	Not applicable
Narrow neural network	Number of fully connected layers: 1; First layer size: 10; Activation: ReLU; Iteration limit: 1000; Regularization strength (Lambda): 0; Standardize data: Yes	19/19	None	Not applicable
Interactions linear	Terms: Interactions; Robust option: Off	19/19	None	Not applicable
Stepwise linear	Initial terms: Linear; Upper bound on terms: Interactions; Maximum number of steps: 1000	19/19	None	Not applicable
Medium Gaussian SVM	Kernel function: Gaussian; Kernel scale: 4.4; Box constraint: Automatic; Epsilon: Auto; Standardize data: Yes	19/19	None	Not applicable
Fine tree	Minimum leaf size: 4; Surrogate decision splits: Off	19/19	None	Not applicable
Quadratic SVM	Kernel function: Quadratic; Kernel scale: Automatic; Box constraint: Automatic; Epsilon: Auto; Standardize data: Yes	19/19	None	Not applicable
Linear	Terms: Linear; Robust option: Off	19/19	None	Not applicable
Robust linear	Terms: Linear; Robust option: On	19/19	None	Not applicable
Linear SVM	Kernel function: Linear; Kernel scale: Automatic; Box constraint: Automatic; Epsilon: Auto; Standardize data: Yes	19/19	None	Not applicable
Fine Gaussian SVM	Kernel function: Gaussian; Kernel scale: 1.1; Box constraint: Automatic; Epsilon: Auto; Standardize data: Yes	19/19	None	Not applicable

Table 3 (continued)

Preset	Hyperparameters	Selected features	Feature ranking algorithm	Optimizer options
Cubic SVM	Kernel function: Cubic; Kernel scale: Automatic; Box constraint: Automatic; Epsilon: Auto; Standardize data: Yes	19/19	None	Not applicable
Custom Gaussian process regression	Signal standard deviation: 0.20121; Optimize numeric parameters: Yes	08/19 NPde, NPk, TC, NPri, NPa, VF, DP, NPcp	ReliefF	Optimizer: Bayesian optimization; Acquisition function: Expected improvement per second plus; Iterations: 30; Training time limit: false
Boosted trees	Minimum leaf size: 8; Number of learners: 30; Learning rate: 0.1; Number of predictors to sample: Select All	19/19	None	Not applicable
Custom Gaussian process regression	Signal standard deviation: 0.20121; Optimize numeric parameters: Yes	15/19 NPk, BFd, DP, BFa, BFv, NPa, BFkv, BFcp, NPd, BFbp, BFst, BFde, BFk, NPri, NPcp	MRMR	Optimizer: Bayesian optimization; Acquisition function: Expected improvement per second plus; Iterations: 30; Training time limit: false
Coarse Gaussian SVM	Kernel function: Gaussian; Kernel scale: 17; Box constraint: Automatic; Epsilon: Auto; Standardize data: Yes	19/19	None	Not applicable
Bagged trees	Minimum leaf size: 8; Number of learners: 30; Number of predictors to sample: Select All	19/19	None	Not applicable
SVM kernel	Learner: SVM; Number of expansion dimensions: Auto; Regularization strength (Lambda): Auto; Kernel scale: Auto; Epsilon: Auto; Iteration limit: 1000	19/19	None	Not applicable
Least squares regression kernel	Learner: Least Squares Kernel; Number of expansion dimensions: Auto; Regularization strength (Lambda): Auto; Kernel scale: Auto; Iteration limit: 1000	19/19	None	Not applicable
Medium tree	Minimum leaf size: 12; Surrogate decision splits: Off	19/19	None	Not applicable
Coarse tree	Minimum leaf size: 36; Surrogate decision splits: Off	19/19	None	Not applicable

(4) Apply statistical methods to select features according to (3) out of all other features.
(5) At the end of steps (3)–(5), you should have a reasonable amount of features and optimal accuracy.

Note that the main focus of this parameter selection is not accuracy but enhanced model learning for generalization. Accuracy is still of utmost importance.

Model evaluation methods

The root mean squared error (RMSE) Eq. (1), mean squared error (MSE) (6), mean absolute error equation (MAE) (7), and the Rsquared equation (R^2) (2)–(5) were applied in this study to measure model performance. The main decision-making performance evaluation metrics in this study was the root mean squared error. This is applied because of its intuitive and direct interpretation of the error.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(h_i - {h_i}^{\text{pred}}\right)^2} \tag{1}$$

$$\overline{h} = \frac{1}{n}\sum_{i=1}^{n} h_i \tag{2}$$

$$\text{SS}_{\text{reg}} = \sum_i \left({h_i}^{\text{pred}} - \overline{h}\right)^2 \tag{3}$$

$$\text{SS}_{\text{tot}} = \sum_i \left(h_i - \overline{h}\right)^2 \tag{4}$$

$$R^2 = \frac{SS_{\text{reg}}}{SS_{\text{tot}}} \tag{5}$$

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(h_i - {h_i}^{\text{pred}}\right)^2 \tag{6}$$

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|h_i - {h_i}^{\text{pred}}\right| \tag{7}$$

Results

See Tables 2 and 3, Fig. 3.

Discussion

The results from Table 2 indicate that the "Custom Gaussian Process Regression" model with the preset "Custom Gaussian Process Regression" performs the best in predicting nanofluid viscosity. This model achieved the lowest RMSE on both the validation and test datasets, indicating its superior predictive accuracy. The other Gaussian Process Regression models also showed promising results but were not as accurate as the top-performing model. The Neural Network models demonstrated competitive performance but were not able to outperform the Gaussian Process Regression models. It is possible that further tuning of the Neural Network architectures and hyperparameters could potentially improve their performance.

The Linear Regression models and Tree-based models showed relatively higher RMSE values, suggesting that they might not capture the complex relationships present in the nanofluid viscosity data as effectively as the Gaussian Process Regression and Neural Network models.

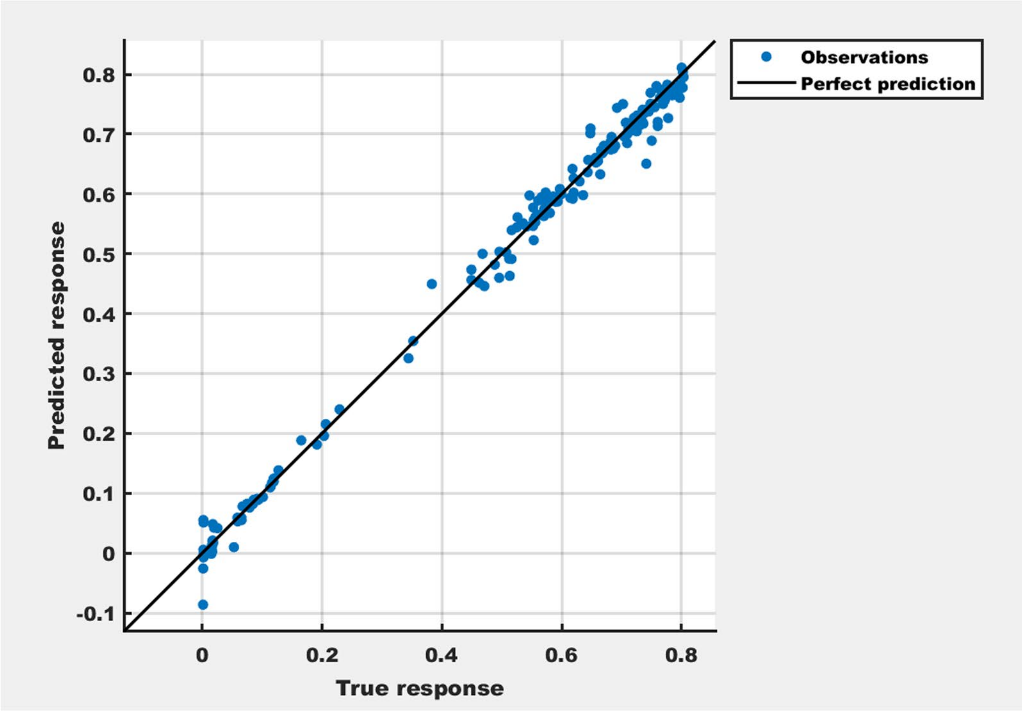
Also, Table 2 shows that the best model is obtained by applying the algorithm in Sect. "Algorithm for parameter selection applied for viscosity" [Novel Feature selection algorithms (NFSA)], with 0.0220 roots mean square of the validation data set. Table 3 presents the settings of each of the models and the applied feature selection algorithm. The model settings in Table 3 were obtained from optimizable version of the original model. They are the settings that give the best results when they are optimized with the Bayesian optimizer class. Table 3 provides insight into the hyperparameters, and feature selection algorithms applied to the Gaussian Process Regression models.

The model with the "None" feature selection algorithm performed well, suggesting that all features can be used essentially for good predictions. However, the "FTest" and "MRMR" feature selection algorithms also showed competitive performance, indicating that they effectively identified relevant features for the nanofluid viscosity prediction.

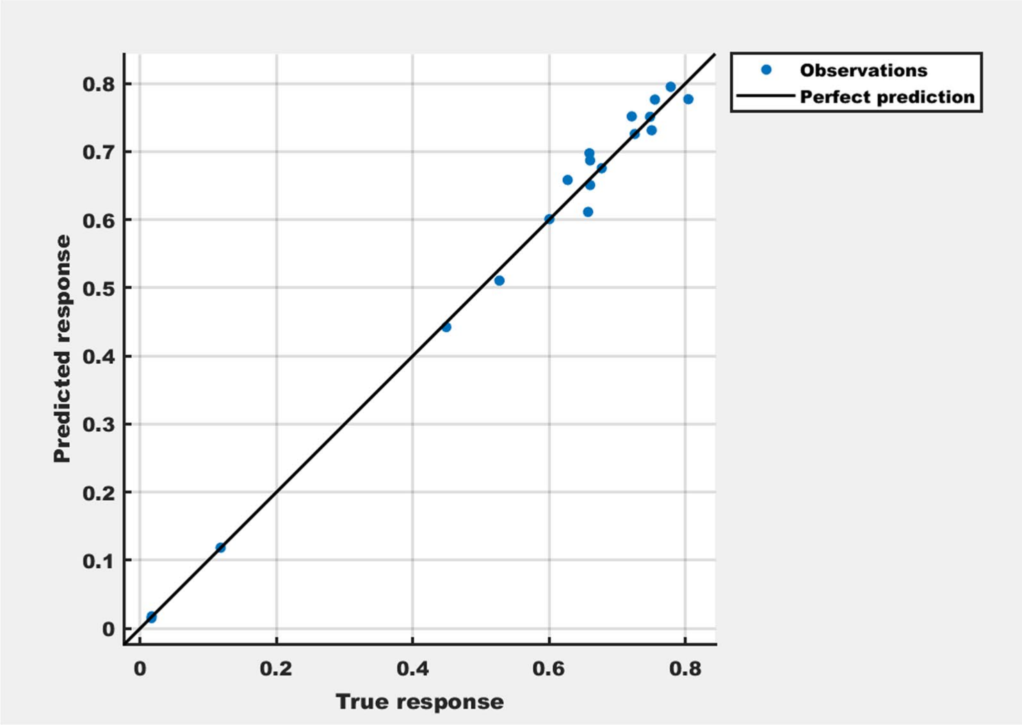
The predictions of response plot of the best model, is shown in Fig. 3. The dots points represent the difference between the predicted response and the true response. A perfect scenario is represented by the line that goes through the origin, indicating that the predicted response and the true response are the same. The vertical distance between the line and any point is the error of the prediction for that point. A good model has small errors, meaning that the predictions are more concentrated near the line.

The plots in Fig. 3a and 3b visually illustrate the quality of predictions made by the accepted model on the training and test datasets, respectively. The close alignment between the predicted responses and the true responses indicates the model's ability to generalize well to unseen data and its overall reliability.

Figures 3a and 3b also demonstrate that the accepted model fits both the training and testing data groups. It is



a A plot of predictions of response in the training data by the accepted model



b A plot of predictions of response in the test data by the accepted model

Fig. 3 **a** A plot of predictions of response in the training data by the accepted model. **b** A plot of predictions of response in the test data by the accepted model essential to note that the models did not have knowledge of the test data during the training process.

It is important to note that the models were not applied to other scenarios due to the marked difference in data logging methods and nanofluid preparation and handling methods by different researchers and considering their good performance on the test data. The future work would be to apply it to other nanofluid thermophysical properties like hybrid nanofluids thermophysical properties which introduce new features that may be important in predicting its thermophysical properties.

Conclusions

This study focused on modeling nanofluid viscosity and optimizing feature selection for accurate prediction. Through the comparison of various models using cross-validation techniques, we gained valuable insights into the factors influencing nanofluid viscosity and identified the most influential features. By incorporating physical meaning into the feature selection process, we achieved improved results. The research findings underscore the importance of considering physical relevance when selecting features for nanofluid viscosity prediction.

By prioritizing features that have a direct physical impact on viscosity, we were able to develop more precise and reliable prediction models. This approach not only enhances the accuracy of viscosity estimation but also provides a better understanding of the underlying mechanisms governing nanofluid behavior.

The application of cross-validation techniques further strengthened our evaluation of the models. By assessing the root mean squared error of the cross-validation sets, we obtained robust measures of model performance. This rigorous evaluation allowed us to identify the most accurate and reliable model for predicting nanofluid viscosity.

The insights gained from this research contribute to the broader understanding of nanofluid viscosity and offer guidance for optimizing their use in practical applications. By accurately predicting viscosity, industries can improve the design and efficiency of heat transfer systems, lubrication processes, and other applications involving nanofluids. The optimized feature selection techniques developed in this study can be readily applied to other nanofluid systems, enabling efficient and effective viscosity prediction models across various applications.

It is important to note that the research presented here focused specifically on single material nanofluids.

Further studies could explore the modeling and feature selection techniques for other types of nanofluids, such as multi-material and hybrid nanofluids or those with complex compositions. Additionally, investigating the relationship between nanofluid viscosity and thermal conductivity could provide valuable insights into the overall fluid behavior.

In conclusion, this study contributes to the field of nanofluid viscosity modeling by providing a novel approach to feature selection and model evaluation. The novel feature selection algorithm makes a more comprehensive method for representing the viscosity of nanofluids in such a way as to preserve the generality of the models. The accurate prediction of nanofluid viscosity opens up new possibilities for optimizing their performance in industrial processes, leading to enhanced efficiency and cost-effectiveness. The models developed in this research serve as valuable tools for predicting nanofluid viscosity and driving advancements in nanofluid-based technologies.

Abbreviations

BFa	Base fluid thermal diffusivity (m^2/s) e+07
BFbp	Base fluid boiling point (°C)
BFcp	Base fluid specific heat capacity (J/(kg K))
BFd	Base fluid density (kg/m^3)
BFde	Base fluid dielectric constant (–)
BFk	Base fluid thermal conductivity (W/(m K))
BFkv	Base fluid kinematic viscosity (m^2/s) e + 07
BFst	Base fluid surface tension (mN/m)
BFv	Base fluid viscosity (Pa·s)
DP	Particle size diameter (nm)
GPR	Gaussian process regressor
MAE	Mean absolute error
ML	Machine learning
MRMR	Minimum redundancy maximum relevance
MSE	Mean squared error
NFSA	Novel feature selection algorithm
NFv	Nanofluid viscosity (Pa·s)
NPa	Nanoparticle thermal diffusivity (m^2/s) e+07
NPcp	Nanoparticle-specific heat capacity (J/(kg K))
NPd	Nanoparticle density (kg/m^3)
NPde	Nanoparticle dielectric constant (–)
NPek	Nanoparticle electrical conductivity (mMS/m)
NPk	Nanoparticle thermal conductivity (W/(m K))
NPmp	Nanoparticle melting point (°C)
NPms	Nanoparticle magnetic susceptibility (–)
NPri	Nanoparticle refractive index (–)
ReLU	Rectified linear unit
RMSE	Root mean squared error
SVM	Support vector machine
TC	Nanofluid temperature (°C)
VF	Volume fraction (%)

Acknowledgements

The author wishes to thank Dr Jongrae Kim and Professor David Barton for their invaluable insights in the preparation of the paper. The author also wishes to thank the Tertiary Education Trust Fund (TET Fund) for funding the studies and the University of Leeds for providing the right environment.

Author contributions

EJO was the main author and only author and carried out all the work in the study.

Funding

TET Fund sponsored the studies at the University of Leeds.

Availability of data and materials

The datasets used and/or analyzed during the current study available from the corresponding author on reasonable request.

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The author declares that there are no competing interests.

Received: 20 July 2023 Accepted: 11 September 2023
Published online: 15 September 2023

References

Aggarwal N, Shukla U, Saxena GJ, Rawat M, Bafila AS, Singh S, Pundir A (2023) Mean based relief: an improved feature selection method based on ReliefF. Appl Intell. https://doi.org/10.1007/s10489-023-04662-w

Bhaumik B, Chaturvedi S, Changdar S, De S (2023) A unique physics-aided deep learning model for predicting viscosity of nanofluids. Int J Comput Methods Eng Sci Mech 24(3):167–181. https://doi.org/10.1080/15502287.2022.2120441

Brownlee J (2016) XGBoost With python: Gradient boosted trees with XGBoost and scikit-learn. Mach Learn Mastery

Buongiorno J, Venerus DC, Prabhat N, McKrell T, Townsend J, Christianson R, Tolmachev YV, Keblinski P, Hu L-W, Alvarado JL (2009) A benchmark study on the thermal conductivity of nanofluids. J Appl Phys 106(9):094312. https://doi.org/10.1063/1.3245330

Çalişkan A (2023) In brain tumor detection, training of mr images created by heat map technique with cnn models-extraction of type-based activation sets and selection of best features by mrmr method. Int Res Eng Sci 5:7

Callister WD (2007) An introduction: material science and engineering. N Y 106:139

Cengel YA, Boles MA, Kanoğlu M (2011) Thermodynamics: an engineering approach, vol 5. McGraw-hill, New York

Chiniforooshan Esfahani I (2023) A data-driven physics-informed neural network for predicting the viscosity of nanofluids. AIP Adv 13(2):025206. https://doi.org/10.1063/5.0132846

Esfandiary M, Mehmandoust B, Karimipour A, Pakravan HA (2016) Natural convection of Al_2O_3–water nanofluid in an inclined enclosure with the effects of slip velocity mechanisms: Brownian motion and thermophoresis phenomenon. Int J Therm Sci 105:137–158. https://doi.org/10.1016/j.ijthermalsci.2016.02.006

Esfe MH, Arani AAA (2018) An experimental determination and accurate prediction of dynamic viscosity of MWCNT (% 40)-SiO_2 (% 60)/5W50 nano-lubricant. J Mol Liq 259:227–237. https://doi.org/10.1016/j.molliq.2018.02.095

Gholizadeh M, Jamei M, Ahmadianfar I, Pourrajab R (2020) Prediction of nanofluids viscosity using random forest (RF) approach. Chemom Intell Lab Syst 201:104010. https://doi.org/10.1016/j.chemolab.2020.104010

Kim JH, Bang IC, Buongiorno J, Venerus DC, Prabhat N, McKrell T, Townsend J, Christianson R, Tolmachev YV, Keblinski P (2009) A benchmark study on the thermal conductivity of nanofluids. J Appl Phys. https://doi.org/10.1063/1.3245330

Lee G-J, Kim CK, Lee MK, Rhee CK, Kim S, Kim C (2012) Thermal conductivity enhancement of ZnO nanofluid using a one-step physical method. Thermochim Acta 542:24–27. https://doi.org/10.1016/j.tca.2012.01.010

Mahesh B (2020) Machine learning algorithms—a review. Int J Sci Res 9(1):381–386. https://doi.org/10.21275/ART20203995

Mathew TE (2023) Breast cancer classification using an extreme gradient boosting model with F-score feature selection technique. J Adv Inf Technol 14(2):363–372

MathWorks T (2022) MATLAB. Version 2022a. In: The Math Works, Inc. www.mathworks.com/

Moran MJ, Shapiro HN, Boettner DD, Bailey MB (2010) Fundamentals of engineering thermodynamics. Wiley, New York

Nguyen C, Desgranges F, Galanis N, Roy G, Maré T, Boucher S, Mintsa HA (2008) Viscosity data for Al_2O_3–water nanofluid—hysteresis: Is heat transfer enhancement using nanofluids reliable? Int J Therm Sci 47(2):103–111. https://doi.org/10.1016/j.ijthermalsci.2007.01.033

Onyiriuka E (2023a) Predictive modelling of thermal conductivity in single-material nanofluids: a novel approach. Preprint. https://doi.org/10.21203/rs.3.rs-3113648/v1

Onyiriuka EJ (2023b) Single phase nanofluid thermal conductivity and viscosity prediction using neural networks and its application in a heated pipe of circular cross section. Heat Transfer 52: 3516–3537. https://doi.org/10.1002/htj.22838

Pastoriza-Gallego M, Casanova C, Páramo R, Barbés B, Legido J, Piñeiro M (2009) A study on stability and thermophysical properties (density and viscosity) of Al_2O_3 in water nanofluid. J Appl Phys 106(6):064301. https://doi.org/10.1063/1.3187732

Pastoriza-Gallego MJ, Casanova C, Legido J, Piñeiro MM (2011) CuO in water nanofluid: influence of particle size and polydispersity on volumetric behaviour and viscosity. Fluid Phase Equilib 300(1–2):188–196. https://doi.org/10.1016/j.fluid.2010.10.015

Rudyak VY, Minakov AV (2018) Thermophysical properties of nanofluids. Eur Phys J E 41(1):15. https://doi.org/10.1140/epje/i2018-11616-9

Said Z, Sundar LS, Rezk H, Nassef AM, Ali HM, Sheikholeslami M (2021) Optimizing density, dynamic viscosity, thermal conductivity and specific heat of a hybrid nanofluid obtained experimentally via ANFIS-based model and modern optimization. J Mol Liq 321:114287. https://doi.org/10.1016/j.molliq.2020.114287

Sakthivel S, Agalya M, Sudha R, Lathika V, Selvi P, Suriyapriya N (2023) Wireless sensor network based anomaly detection using SVM-RFE-MRMR. In: 2023 7th international conference on intelligent computing and control systems (ICICCS)

Sarker IH (2021) Machine learning: algorithms, real-world applications and research directions. SN Comput Sci 2(3):160. https://doi.org/10.1007/s42979-021-00592-x

Tan KX, Ilyas SU, Pendyala R, Shamsuddin MR (2022) Assessment of thermal conductivity and viscosity of alumina-based engine coolant nanofluids using random forest approach. AIP Conf Proc. https://doi.org/10.1063/50099553

Tavman I, Turgut A, Chirtoc M, Schuchmann H, Tavman S (2008) Experimental investigation of viscosity and thermal conductivity of suspensions containing nanosized ceramic particles. Arch Mater Sci 100(100):99–104

Tm P, Veni S (2023) Hybrid feature selection model based on rfe and mrmr on anxiety disorder dataset. J Theor Appl Inf Technol 101(10)

Venerus DC, Buongiorno J, Christianson R, Townsend J, Bang IC, Chen G, Chung SJ, Chyu M, Chen H, Ding Y (2010) Viscosity measurements on colloidal dispersions (nanofluids) for heat transfer applications. Appl Rheol. https://doi.org/10.3933/applrheol-20-44582

Venkatesan S (2023) Design an intrusion detection system based on feature selection using ML algorithms. Math Stat Eng Appl 72(1):702–710

Yadav D, Dansena P, Ghosh SK, Singh PK (2020) A unique multilayer perceptron model (ANN) for different oxide/EG nanofluid's viscosity from the experimental study. Phys Stat Mech Appl 549:124030. https://doi.org/10.1016/j.physa.2019.124030

Publisher's Note

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