Onyiriuka
Bulletin of the National Research Centre (2023) 47:140
https://doi.org/10.1186/s42269-023-01115-9

Bulletin of the National Research Centre

RESEARCH

Open Access

Predictive modelling of thermal conductivity in single-material nanofluids: a novel approach

Ekene Onyiriuka[1*]

Abstract

Background This research introduces a novel approach for modelling single-material nanofluids, considering the constituents and characteristics of the fluids under investigation. The primary focus of this study was to develop models for predicting the thermal conductivity of nanofluids using a range of machine learning algorithms, including ensembles, trees, neural networks, linear regression, Gaussian process regressors, and support vector machines.

The main body of the abstract To identify the most relevant features for accurate thermal conductivity prediction, the study compared the performance of established feature selection algorithms, such as minimum redundancy maximum relevance, Ftest, and RReliefF, a newly proposed feature selection algorithm. The novel algorithm eliminated features lacking direct implications for fluid thermal conductivity. The selected features included temperature as a thermal property of the fluid itself, multiphase features such as volume fraction and particle size, and material features including nanoparticle material and base fluid material, which could be fixed based on any two intensive properties. Statistical methods were employed to select the features accordingly.

Results The results demonstrated that the novel feature selection algorithm outperformed the established approaches in predicting the thermal conductivity of nanofluids. The models were evaluated using fivefold cross-validation, and the best model was the model based on the proposed feature selection algorithm that exhibited a root-mean-squared error of validation of 1.83 and an R-squared value of 0.94 on validation set. The model achieved a root-mean-squared error of 1.46 and an R-squared value of 0.97 for the test set.

Conclusions The developed predictive model holds practical significance by enabling nanofluids' numerical study and optimisation before their creation. This model facilitates the customisation of conventional fluids to attain desired fluid properties, particularly their thermal properties. Additionally, the model permits the exploration of numerous nanofluid variations based on permutations of their features. Consequently, this research contributes valuable insights to the design and optimisation of nanofluid systems, advancing our understanding and application of thermal conductivity in nanofluids and introducing a novel and methodological approach for feature selection in machine learning.

Keywords Single material, Nanofluids, Modelling, Predict, Thermal conductivity, Feature selection

Background

This research introduces a novel method for modelling nanofluid thermophysical properties (thermal conductivity of single-material nanofluid). It uses the physics of the fluid to select its features. Using this approach ensures a generalised physical model. The implication of such an approach is creating a model that meets the needs of many cases of single-material nanofluids. This is so because the feature selection was physics-based.

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

This approach is unusual as much research depends on statistical tools to select its learning features (MathWorks 2022).

Literature review

The prediction of the thermal conductivity of nanofluids has been studied extensively. The following reviews give the state of the art on this topic as given by various researchers, beginning with some historical studies to present works.

Xie et al. (2002) studied the thermal conductivity measurement of SiC suspension in water and ethylene glycol and the effect of the size and shape of the added solid phase on the enhancement of thermal conductivity. Experimental data for SiC nanoparticles in water and ethylene glycol were presented. The thermal conductivity of SiC nanofluid was measured using a transient hot-wire method. The effects of the morphologies (size and shape) of the added solid phase on the enhancement of the thermal conductivity of the nanoparticle suspension were studied. This study was one of the first to supply such data. Furthermore, it was one of the first to report the effects of morphology on thermal conductivity enhancement. It highlighted the deviation in the existing Hamilton–Crosser model with spherical and cylindrical assumptions. The study considers just one type of nanoparticle (SiC). However, only two particle sizes were considered. In the study, it was assumed that heat transfer between the particles and fluid takes place at the particle–surface interface. Heat transfer is expected to be more efficient and rapid for a system with a larger interfacial area. As the particle sizes decrease, the effective thermal conductivity of the suspension improves. Higher thermal conductivities were obtained by adding SiC nanoparticles. Furthermore, it was observed in the study that a linear relationship existed between low volume fraction in the (1–5%) volume fraction range and the thermal conductivity enhancements.

Murshed et al. (2005) studied the thermal conductivity of TiO_2 water nanofluid in their paper. A more convenient measurement of the thermal conductivity of nanofluids was created—A transient hot-wire apparatus with an integrated correlation model. A relationship was established between particle volume fraction, shape, and thermal conductivity. The study focused on conveniently measuring nanofluids' thermal conductivity and comparing results with the theoretical prediction. The study was one of the first to collect and compare such data with theoretical models. However, only one type of nanofluid was used. They pointed out that traditional models fail due to a lack of accounting of the effects of: (1) particle size, (2) particle Brownian motion, (3) nano-layering, (4) effects of nanoparticle clustering—an integrated correlation model allowed for a more precise and convenient measurement of the thermal conductivity of nanofluids. Further efforts to develop a suitable model to predict the thermal conductivity of nanofluids will consider other factors that are important in enhancing the heat transfer performance of nanofluids.

Komeilibirjandi et al. (2020) studied the thermal conductivity of nanofluids containing two nanoparticles and predicted it by using correlation and artificial neural network. The GMDH (Group method of data handling) Neural network was applied to model the thermal conductivity of CuO–nanofluids. Water and ethylene glycol were the base fluids. % volume fraction, nanoparticle size, temperature, and thermal conductivity of the base fluid were considered. Data used were extracted from experimental studies in the literature.

It is worth knowing that most researchers, as outlined above, have attempted this modelling. Furthermore, researchers that attempt the generalised model form have fixed their models to only the nanofluid types collected. Implying no other nanofluid type outside their collected data can be accounted for.

Ramezanizadeh et al. (2019) mentioned the two types of nanofluids: conventional or single-material nanofluids and hybrid nanofluids. They reviewed proposed models for predicting the thermal conductivity of various researchers. The following conclusions can be drawn from their report (Ramezanizadeh et al. 2019):

(a) The reviewed models were not tested with out-of-sample data.
(b) Many models were made for a specific nanofluid (meaning they only covered one nanoparticle and base fluid type). The percentage of such models was 89% (23) out of all 26 models reviewed.
(c) The remaining three (3) models were designed to cover more than one nanofluid. However, they were limited in the number of nanofluids on which they could make predictions due to the numerous nanofluid types that exist in the literature plus those that can be fabricated. A further study of the modelling approach used by these researchers reveals that a shift in convention in the choice of model features might solve this problem. For example, one researcher Ahmadloo and Azizi (2016) considered numbers that would differentiate each nanoparticle type and base fluid. Although this helped add a distinct factor to the conventional inputs of particle size, volume fraction, and temperature, the resulting model was still limited to 15 nanofluid types and could not be applied beyond those nanofluids. Also, adding ordinate numbers as opposed to encoding (one-hot types) has been shown in machine learn-

ing to bias models by making those with higher numbers more critical.

(d) For the final two models of the three in (c). They could not distinguish between nanofluid types due to the features they selected, so they were only accurate in a limited range of parameters and thus not useful outside of those ranges.

(e) The models' focus was curve fitting, not prediction.

The other group of models studied by Ramezanizadeh et al. (2019) are the correlation types with low accuracy and a narrow range where they hold; hence, they are usually avoided.

In this study, the predictors used as input were chosen so that they uniquely represented the nanoparticle and base fluid data and could also apply to other nanoparticles and base fluids not available in the collected data. This ensures that it can be used to make predictions based on the numerical values of the predictors only and hence be a more general model. As compared to the work of other researchers such as Ahmadloo and Azizi (2016), as mentioned above, used predictors that were only uniquely identified with the nanoparticle and base fluid in the collected data; hence, they could not be used on a general basis for predicting the thermal conductivity of single material nanofluids not included in the collected data. Moreover, our approach in this study is to create a generalised model that accounts for all single-material nanofluids using a novel feature selection algorithm.

Experimental measurements and description of the experimental setup and procedures

Data used in this study were obtained from experimental data reported in the following articles (Patel et al. 2010):

The report's experimental set-up for measuring thermal conductivity utilised a transient hot-wire apparatus. The measurement cell consisted of a 15-cm-long platinum wire with a diameter of 100 μm. The platinum wire served both as a heater and a thermometer. It was placed in a glass container filled with the test liquid and formed an arm of a Wheatstone bridge. An analytical solution for the temperature distribution was employed to determine the thermal conductivity of the test liquid. This solution assumes an infinitely long line heat source continuously heating a semi-infinite medium. The platinum wire was electrically insulated to prevent interference. The validity of the measurement technique was established by comparing the obtained thermal conductivity values with literature values for various fluids such as water, ethylene glycol, transformer oil, xylene, and toluene. The results showed that the measurements obtained from the transient hot-wire apparatus were within 1.2% of the literature values, indicating its reliability. However, it should be noted that this equipment is not suitable for measuring the thermal conductivity of fluids with high electrical conductivities. Nonetheless, it proved effective for measuring the thermal conductivity of oxide nanofluids, which was the focus of their study. Overall, the transient hot-wire equipment employed in the study provided a robust and validated method for measuring thermal conductivity, ensuring accurate and reliable data for the analysis of nanofluids.

Machine learning techniques

Machine learning (ML) techniques (Ewim et al. 2020, 2021; Géron 2022; Jiang et al. 2020; Meng et al. 2020; Sharma et al. 2022; Zhu et al. 2021) have revolutionised regression analysis by providing powerful tools for predicting continuous numerical outcomes. This section will explore several ML regression techniques commonly used in various domains. These techniques include neural networks, gradient boosting, random forest, support vector machine (SVM), linear models, decision trees, and naive Bayes regression models. Moreover, they have been investigated for nanofluid thermal conductivity predictions in this study along with the application of the novel feature selection algorithm proposed by this study.

Neural networks

Neural networks are ML models inspired by the human brain's neural structure. They consist of interconnected layers of artificial neurons that can learn complex patterns and relationships. Neural networks have been successfully applied to regression tasks because they capture nonlinear relationships in the data (Chiniforooshan Esfahani 2023; Genzel et al. 2022; Hornik et al. 1989; Kamsuwan et al. 2023; Kannaiyan et al. 2019; Mijwil 2018; Ekene Jude Onyiriuka 2023a, b; Peng et al. 2020).

Gradient boosting

Gradient boosting is an ensemble learning method that combines multiple weak models, typically decision trees, to create a robust predictive model. It trains new models to correct the errors made by previous models, gradually improving the overall prediction accuracy (Friedman 2001).

Random forest

Random forest is another ensemble learning technique that constructs a collection of decision trees and combines their predictions to make accurate predictions. It reduces overfitting by introducing randomness in tree-building (Breiman 2001; Gholizadeh et al. 2020; Tan et al. 2022).

Support vector machine (SVM)

SVM is a popular ML algorithm used for regression tasks. It aims to find the best hyperplane that separates the data into different classes while minimising the error in the training instances. SVM can handle linear and nonlinear regression problems (Razavi et al. 2019; Vapnik 1999).

Linear model

Linear regression is a simple and widely used ML technique for regression analysis. It assumes a linear relationship between the input features and the target variable. The goal is to find the best-fit line that minimises the sum of squared differences between the predicted and actual values (Géron 2022).

Decision trees

Decision trees are versatile ML models that make predictions by partitioning the feature space into regions based on simple decision rules. They are interpretable and can capture nonlinear relationships in the data. Decision trees can be used for regression and classification tasks (Breiman et al. 1984).

Naive Bayes regression model

Naive Bayes regression is based on Bayes' theorem and assumes that the input features are conditionally independent given the target variable. Despite its simplicity, naive Bayes can provide reasonable predictions, especially when the independence assumption holds (Rish 2001).

These ML regression techniques offer various options for analysing and predicting continuous variables. The choice of technique depends on the specific problem, dataset characteristics, interpretability requirements, and computational considerations. Researchers and practitioners often compare and combine these techniques to achieve the best performance for their regression tasks (Sharma et al. 2022; Witten et al. 2016; Yashawantha and Vinod 2021; Zhu et al. 2021).

Cross-validation

Cross-validation is a machine learning technique used to estimate the ability of a machine learning model on unknown data. In this process, a small sample is used to assess how the model will perform. These data are called "out-of-bag" data. It is a popular strategy since it is simple and produces a less biased or optimistic estimate of a model's predictive ability than other processes, such as a simple train–test split. Cross-validation ensures a fair comparison of the models (Brownlee 2020a).

To compare machine learning algorithms well, we ensured that each algorithm was evaluated on the same data and in the same way. Cross-validation is a resampling technique for evaluating machine learning models on a small sample of data. Although there are several cross-validation methods, we chose the k-fold cross-validation method because it was the most fitting for this study—for evaluating models without bias. However, researchers Brownlee (2020b) reported an alternative method called stratified cross-validation, which is suitable for cross-validating imbalanced datasets. However, it is only applicable to classification problems. Hence, it does not apply to our study. The process of k-fold cross-validation includes a parameter, k, which specifies the number of groups into which a given data sample should be sectioned. Fivefold cross-validation refers to cross-validation where the dataset is split into five sections. The cross-validation method used in this study was the fivefold cross-validation method, which is applied to evaluate every algorithm to ensure the same evaluation on all models.

The figure below shows the k-fold algorithm.

The general procedure is shown in Fig. 1:

1. Randomised shuffling of the dataset.
2. Divide the dataset into k distinct sections.
3. For each distinct section:

    i. Use the section as a test data set.
    ii. Use the remainder of the section as a training data set.

Iteration 01	Test	Train	Train	Train	Train
Iteration 02	Train	Test	Train	Train	Train
Iteration 03	Train	Train	Test	Train	Train
Iteration 04	Train	Train	Train	Test	Train
Iteration 05	Train	Train	Train	Train	Test

Fig. 1 The fivefold algorithm

iii. Fit each model to the training set (ii) and evaluate it on the test set (i).
iv. Save the evaluation score and note the model.

4. Steps (i) through (iv) should be repeated for x number of models.
5. Report the predictive ability of each model by summarising the model's average evaluation score.

Evaluation metrics

The evaluation metrics are used to measure the performance of the predictive model. Standard metrics for regression tasks include:

Root-mean-squared (RMSE) error (Hastie et al. 2009): it is a popular evaluation metric for regression tasks. It is derived from the mean-squared error (MSE) as seen in Eq. 1 by taking the square root of the average of the squared differences between the predicted values and the actual values. The rationale behind using RMSE as an evaluation metric is like that of MSE, but with some additional considerations RMSE, like MSE, provides a measure of the average prediction error. However, by taking the square root of MSE, RMSE is expressed in the same units as the target variable, making it more interpretable and easier to relate to the original scale of the problem. For example, if the target variable represents the temperature of a fluid in °C, RMSE will also be expressed in °C. Like MSE, RMSE also emphasises more significant errors by squaring them. However, by taking the square root of MSE, RMSE balances penalises significant errors and maintaining interpretability. It allows a more intuitive understanding of the typical magnitude of errors in the model's predictions. RMSE enables direct comparison of models or different scenarios, as it provides a scale-dependent standard metric. When comparing models, lower RMSE values indicate better performance, indicating that the model's predictions are closer to average. RMSE is related to the standard deviation of the errors. It measures the typical spread or dispersion of the errors around the actual values. Smaller RMSE values suggest a more concentrated distribution of errors, indicating better accuracy and precision in the model's predictions.

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

RSQUARED(Gareth et al. 2013) is a widely used evaluation metric for regression tasks. It measures the proportion of the variance in the dependent variable that is predictable from the independent variables as seen in Eqs. (2)–(5). RSQUARED measures how well the regression model fits the observed data. It quantifies the proportion of the total variation in the dependent variable that the independent variables can explain. Higher RSQUARED values indicate a better fit and suggest that the model accounts for a more significant proportion of the variation in the target variable. RSQUARED allows for comparison against a baseline model, often the mean of the dependent variable. An RSQUARED value of 1 indicates that the model perfectly predicts the target variable, while a value of 0 suggests that the model does not provide any improvement over the baseline model. RSQUARED has a straightforward interpretation as the proportion of variance explained. It provides a convenient measure to communicate the model's predictive power to non-technical researchers, such as decision-makers.

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

$$SS_{reg} = \sum_{i} (h_i{}^{pred} - \overline{h})^2 \tag{3}$$

$$SS_{tot} = \sum_{i} \left(h_i - \overline{h}\right)^2 \tag{4}$$

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

Mean-squared error (MSE) (Hastie et al. 2009) is a commonly used evaluation metric for regression tasks. It measures the average squared difference between predicted and actual values as shown in Eq. (6). MSE measures the average prediction error by considering the squared differences between the predicted and actual values. It penalises more significant errors due to the squaring operation, providing a way to assess the accuracy of the model's predictions. MSE is mathematically convenient and has desirable properties for optimisation. It is differentiable, allowing efficient gradient-based optimisation algorithms commonly used in machine learning. This property makes MSE practical for training regression models using gradient-based optimisation techniques. MSE can be decomposed into the sum of variance and bias terms, known as the bias–variance trade-off. This decomposition provides insights into the model's performance by assessing the balance between overfitting (low bias, high variance) and underfitting (high bias, low variance). By minimising MSE, the model aims to find the optimal balance between bias and variance for improved generalisation.

Differentiable and Mathematically Convenient:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(h_i - h_i^{pred})^2 \quad (6)$$

Mean absolute error (MAE) (Gareth et al. 2013) is a commonly used evaluation metric for regression tasks. It measures the average absolute difference between predicted and actual values as seen in Eq. (7).

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|h_i - h_i^{pred}\right| \quad (7)$$

MAE is less sensitive to outliers compared to other error metrics like MSE. Since MAE calculates the absolute difference, it does not heavily penalise significant errors. This makes MAE more robust to outliers, minimising their influence on the overall error measurement. MAE has a straightforward interpretation as the average absolute difference between the predicted and actual values. It represents the average magnitude of the errors, providing an intuitive understanding of the model's performance. MAE is easily understandable by non-technical researchers, making it suitable for communication. MAE is mathematically simple and computationally efficient. It does not involve squaring the differences, simplifying calculations and reducing the computational complexity of evaluating the error metric.

Methods

The procedure to evaluate the predictive model involves the following: The trained model was used to make predictions on the validation dataset. Moreover, the above evaluation metrics were computed using the predicted values and the corresponding actual values from the validation dataset.

The calculated metrics are presented in Table 2. The validation RMSE is used to sort the table from best to worse.

Data exploration and models

The variables were collected for analysis, selecting which features might be necessary for modelling the thermal conductivity of nanofluids.

The nanofluid data set was collected from experimental studies. It consisted of 348 data points.

Conceptualisation and parameter selection

Conceptualisation in this context is formulating a novel parameter selection method for predicting the thermal conductivity of all single-material nanofluids. Parameter selection is the selection of the training data characteristics learned during the learning process. In this study, there is a shift from parameter selection based solely on statistics to selection based on physics and reduction of an initially large dimensional space. It describes a feature engineering technique where all variables possible are selected to increase the dimensional space of a dataset by increasing the number of descriptive features. The goal is to find an optimal physical dimensional set of features that make each data point distinct from the others. Even though these extra variables may not have high importance in predicting the target variables, their presence in the model helps to make the predictions unique. This is a different approach from conventional feature engineering. In conventional feature engineering, feature engineering aims to create new variables from existing ones to improve the performance of a machine learning model by providing more information to the model (Patel 2021).

In this case, the new variables are not derived from the existing variables but are other independent variables that further define the characteristics of what is being predicted.

It is important to remember that feature engineering is an iterative process that necessitates a thorough understanding of the problem and the data at hand.

Proposed algorithm for parameter selection

Here we discuss the procedure for selecting parameters according to the novel method discussed above to predict the thermal conductivity of all single-material nanofluids.

(1) Check the problem being solved.
(2) List all the possible features (start with the largest number of features/dimensional space possible).
(3) Manually drop features that have no meaning or direct implication to the thermal conductivity 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.

(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 number of features and optimal accuracy.

Note that this parameter selection focuses on accuracy and enhanced model learning for generalisation. Accuracy is still of utmost importance.

Other feature selection algorithm

The section presents the minimum redundancy maximum relevance (MRMR) and RReliefF.

Minimum redundancy maximum relevance (MRMR).

- Begin by picking the most relevant feature from a set and add it to a selected set (S).
- Check the remaining features (Sc) for those with relevant information but not redundant with the ones in S.
- If such features exist, add the most relevant of them to S.
- Keep doing this until there are no more non-redundant features left in Sc.
- Find the Sc feature with the highest value considering its ability to contribute new information—Mutual Information Quotient (MIQ) while balancing relevance and redundancy.
- Add this feature to S and repeat Step 4 as needed.
- Include Sc features with zero relevance into S, randomly.

The algorithm chooses features that are informative and distinct, resulting in an optimised subset for analysis.

RReliefF.

Initialise the weights W_{dy}, W_{dj}, $W_{dy\wedge dj}$, and W_j to 0.

The algorithm then follows these steps for a certain number of iterations, denoted by 'updates.'

For each iteration 'i' and for a randomly chosen observation x_r

Find the k-nearest observations to x_r.

m is the number of iterations specified by `'updates'`.

Update the intermediate weights as follows:

$$W_{dy}^{i} = W_{dy}^{i-1} + \Delta_y(x_r, x_q) * d_{rq}$$

$$W_{dj}^{i} = W_{dj}^{i-1} + \Delta_j(x_r, x_q) * d_{rq}$$

$$W_{dy\wedge dj}^{i} = W_{dy\wedge dj}^{i-1} + \Delta_y(x_r, x_q) * \Delta_j(x_r, x_q) * d_{rq}$$

The $\Delta_y(x_r, x_q)$ calculates the difference in continuous response y between observations x_r and x_q normalised by the range of response values:

$$\Delta_y(x_r, x_q) = \frac{|y_r - y_q|}{\max(y) - \min(y)}$$

where y_r is the response value for observation x_r; y_q is the response value for observation x_q. d_{rq} is a distance function.

After updating all intermediate weights for each iteration, RReliefF calculates the predictor weights W_j using the formula:

$$W_j = \frac{W_{dy\wedge dj}}{W_{dy}} - \frac{W_{dj} - W_{dy\wedge dj}}{m - W_{dy}}$$

Preprocessing of experimental data for training and validation

Preprocessing experimental data for training and validation involve several steps to ensure the data are in a suitable format and quality for ML regression analysis. The following are essential components of data preprocessing for training and validation: The removal of any irrelevant or redundant data did not contribute to the regression task. This includes removing duplicates, handling missing values, and addressing outliers. Missing values can be imputed using techniques such as mean, median, or advanced imputation methods like regression imputation. However, this study did not impute missing values since the data had no missing values.

Feature selection and modelling

In this study, the modelling process is approached from the standpoint of feature selection.

To start the modelling procedure, we first designed it for reproducibility. This was achieved by using a default and consistent random seed generator. The data are then partitioned into two sets in an 80:20 ratio, 80% for training (252 observations) and 20% (69 observations) for later out-of-bag testing. And 10% of the 80% for testing (27 observations). The fivefold cross-validation was carried out to select the model without bias fairly.

The response (percentage enhancement of thermal conductivity—"ENT") was specified. Moreover, the rest of the variable was specified as the predictors.

Results

Discussion

Data analysis

The total number of data rows collected was 348, with 22 columns including the response variable.

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

Figure 2 showcases histograms portraying the characteristics of each variable, including the response variable "ENT" response variable. Visual scrutiny of these histograms swiftly indicates that none of the variables conforms to a normal distribution, prompting the

Table 1 Common feature selection algorithm

Feature selection type	Brief description of the feature selection algorithm	Selected features and their importance	Model performance	
Minimum redundancy maximum relevance (MRMR)	The MRMR (minimum redundancy maximum relevance) algorithm aims to identify an optimal feature subset highly relevant to the response variable and maximally dissimilar	VF = 0.2279, TC = 0.2000	RMSE (Validation)	5.65
			MSE (Validation)	31.96
			RSQUARED (Validation)	0.40
			MAE (Validation)	4.57
			MAE (Test)	5.34
			MSE (Test)	39.61
			RMSE (Test)	6.29
			RSQUARED (Test)	0.25
FTest (Importance > 25)	This involves conducting separate Chi-square tests for each predictor variable to determine if there is a significant association between the predictor and the response	VF = 50.7728, DP = 31.8726, NPk = NPa = NPcp = NPmp = NPri = NPek = NPms = NPd = 26.5023	RMSE (Validation)	3.50
			MSE (Validation)	12.27
			RSQUARED (Validation)	0.78
			MAE (Validation)	2.64
			MAE (Test)	2.33
			MSE (Test)	9.72
			RMSE (Test)	3.12
			RSQUARED (Test)	0.77
RReliefF (> Abs (0.01))	The RReliefF algorithm considers the consistency of predictor values among neighbours with the same response values. It penalises predictors exhibiting inconsistent values among neighbouring instances with the same response while rewarding predictors demonstrating differing values among neighbours with different response values	VF = 0.1515, BFv = 0.0142, BFkv = 0.0126, DP = − 0.0092	RMSE (Validation)	2.66
			MSE (Validation)	7.07
			RSQUARED (Validation)	0.86
			MAE (Validation)	1.94
			MAE (Test)	1.95
			MSE (Test)	7.39
			RMSE (Test)	2.72
			RSQUARED (Test)	0.90

requirement for models attuned to handling such non-normal data distributions. Therefore, a range of modelling methodologies comes into view. Notably, robust linear regression (Maronna et al. 2019) emerges as a promising method, particularly due to its reliable coefficient estimates even in the presence of outliers. Likewise, nonparametric models, including decision trees, random forests, and support vector machines (Kurani et al. 2023), surface as possible modelling options, capable of understanding intricate relationships without making such assumptions about data distribution. Furthermore, ensemble models, represented by boosting and bagging (Mohammed & Kora 2023), also exhibit their strength to enhance predictive accuracy. This proves invaluable when dealing with non-normal data and intricate nonlinear relationships. Additionally, the potency of neural networks (Cong & Zhou 2023) becomes evident, owing to their remarkable capacity for detecting patterns and relationships even amidst complex data representation. It is noteworthy that the author avoids data transformation of the response variable since that could potentially lead to data leakage because data transformation holds the risk of inadvertent data leakage as noted by Osborne (2010) where knowledge from the target variable infiltrates the transformation process, affecting the model outcomes. In conclusion, Fig. 2 effectively visualises the histogram plots of diverse variables, exposing their departure from normality. Consequently, a suite of modelling options is proposed, encompassing robust linear regression, nonparametric models (e.g., decision trees, random forests, support vector machines), ensemble models (e.g., boosting, bagging), and neural networks. These selections are apt for handling data exhibiting non-normal distributions. The selection among these modelling approaches should be guided by the specific data characteristics.

Figure 3b presents the box plot for each variable, offering a visual summary of their distribution characteristics, including skewness, symmetry, and potential outliers. The structure of the box plot is depicted in Fig. 3a. Box plots allow for a concise representation of multiple variables, facilitating the identification of differences in central tendency and variability among them. The box plot provides several important features for each variable. The rectangular box represents the interquartile range (IQR), encompassing the middle 50% of the data. The line within the box represents the median, indicating the central

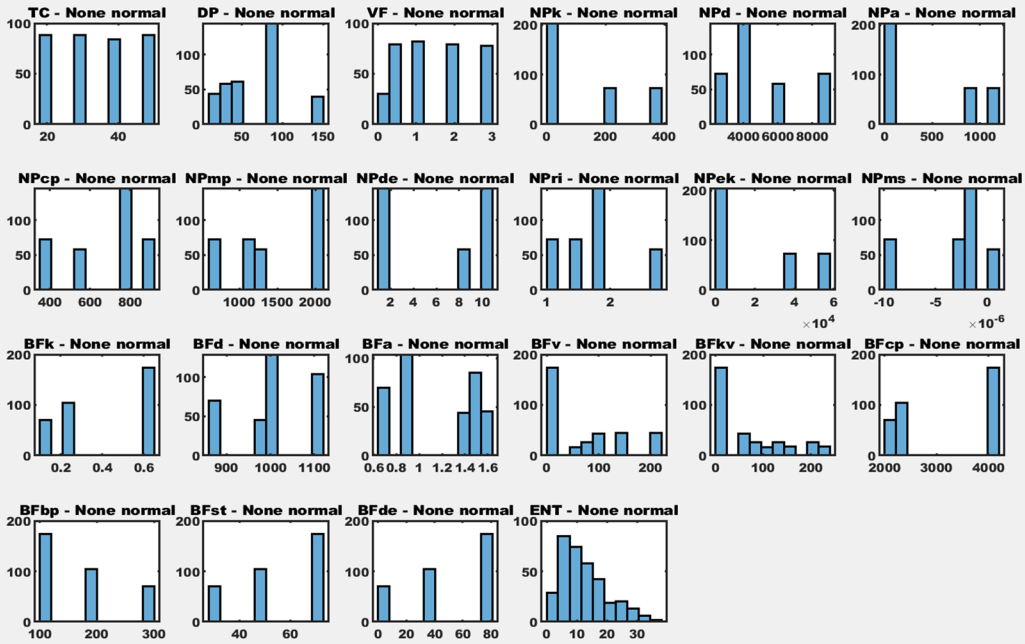

Fig. 2 A histogram plot of each variable

tendency of the variable. The whiskers extend from the box, indicating the data range within 1.5 times the IQR. Values beyond the whiskers are considered potential outliers and are displayed as individual data points. By examining the box plots in Fig. 3b, we can observe the distributional characteristics of each variable. Skewness can be observed by considering the asymmetry of the box and whiskers. The whisker lengths are significantly different, suggesting unequal variability. Outliers are visually identifiable as can be observed by data points lying beyond the whiskers. The side-by-side presentation of multiple variables in Fig. 3b allows for an easy comparison of their central tendencies and variabilities. Differences in box lengths, medians, and whisker lengths among the variables indicate variations in their distributions. The utilisation of box plots aids in understanding the distributional properties of each variable and enables the identification of potential outliers and variations in central tendency and variability across multiple variables. We can observe similar data characteristics between the variables regarding skewness, making it possible to create some groupings. In contrast, some variables are single and do not fall under similarity groupings. The following groups 1 to 12 show the variables that have similar relationships with themselves by visual examination as observed in the box plot in Fig. 3.

Group 1: Nanofluid temperature
Group 2: Particle size diameter, Nanoparticle density, Nanoparticle thermal conductivity
Group 3: Volume fraction, Base fluid surface tension
Group 4: Nanoparticle thermal diffusivity, Base fluid dielectric constant
Group 5: Nanoparticle-specific heat capacity, Nanoparticle electrical conductivity
Group 6: Nanoparticle melting point, Base fluid specific heat capacity
Group 7: Nanoparticle dielectric constant, Base fluid density
Group 8: Nanoparticle refractive index, Base fluid boiling point
Group 9: Nanoparticle magnetic susceptibility
Group 10: Base fluid thermal conductivity, Base fluid thermal diffusivity
Group 11: Base fluid viscosity
Group 12: Base fluid kinematic viscosity, Percentage enhancement of nanofluid thermal conductivity

Groups 1, 9, and 11 are very different from the rest.

Results analysis

In the appendix, the table presents the results of the model selection process. The neural networks emerged

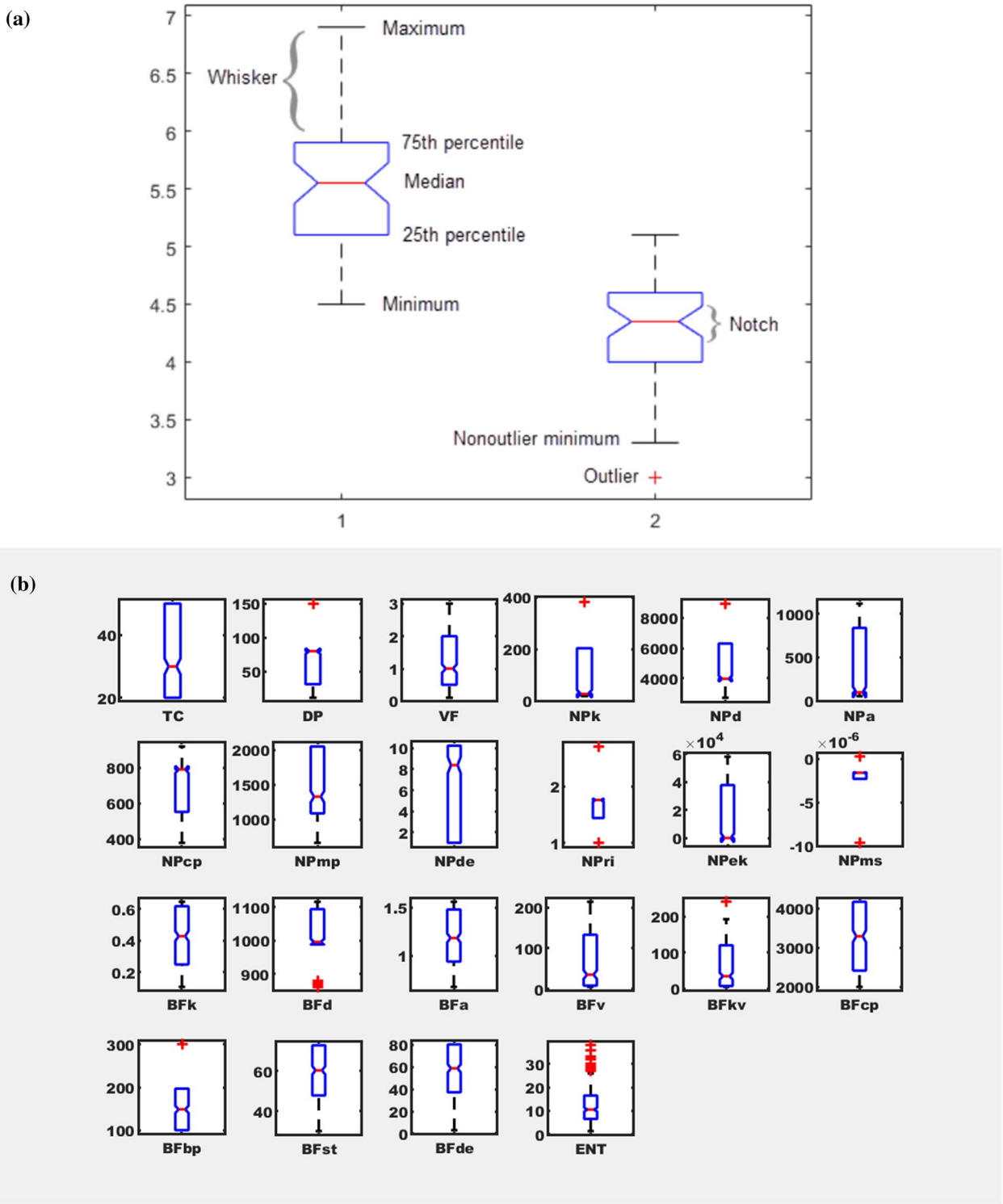

Fig. 3 **a** Box plot anatomy (MathWorks 2022). **b** Box plot of the variables

as the best basic model based on the cross-validated set's root-mean-squared error (RMSE). The selected neural network consisted of three fully connected layers with sizes of 10, 10, and 10, respectively. The rectified linear unit (ReLU) was used as the activation function, and the regularisation strength was set to zero. The

implementation of the neural network had a variable learning rate and a validation check stopping criteria. Additionally, the data were standardised before training the model. To further optimise the neural network, Bayesian optimisation was employed. The optimisation process was guided by the estimated improvement per second plus 30 iterations. The hyperparameter search range included the number of fully connected layers (1–3), the size of the first layer (1–300), the size of the second layer (1–300), the size of the third layer (1–300), and the choice of the activation function (ReLU, tanh, sigmoid, or none). The regularisation strength varied between 3.9683×10^(-08) and 396.8254. Data standardisation was considered a binary choice (yes/no). The optimised hyperparameters for the neural network were determined as follows: two fully connected layers with sizes of 64 and 10, respectively. The ReLU activation function was utilised, the regularisation strength was set to 392.6291, and the data were standardised. However, it was observed that the performance of the optimised neural network was not satisfactory. The validation RMSE for the optimised model was 14.472, which was higher than the non-optimised version. The mean-squared error (MSE) was 209.443, the R-squared (RSQUARED) was -2.841, and the mean absolute error (MAE) was 12.482. For the test set consisting of 27 observations, the MAE was 9.096, the MSE was 125.532, the root-mean-squared error (RMSE) was 11.204, and the RSQUARED was -1.932. This observation is possibly due to overfitting the model parameters to the data and hence the poor performance on the validation set and test set. In order to perform feature selection, a copy of the most accurate model was used. Table 1 presents the results of the analysis using standard feature selection algorithms. The minimum redundancy maximum relevance (MRMR) algorithm identified VF and TC as the selected features with respective importance scores of 0.2279 and 0.2000. The model's performance based on validation data included an RMSE of 5.65, an MSE of 31.96, an RSQUARED of 0.40, and an MAE of 4.57. For the test set, the MAE was 5.34, the MSE was 39.61, the RMSE was 6.29, and the RSQUARED was 0.25. Another feature selection algorithm, FTest, was employed with an importance threshold of 25. This approach involved conducting separate Chi-square tests for each predictor variable to determine their significant association with the response. The resulting selected features and their importance scores were VF (50.7728), DP (31.8726), and others with importance scores of 26.5023. The model's performance improved compared to the MRMR-selected features, with an RMSE of 3.50, an MSE of 12.27, an RSQUARED of 0.78, and an MAE of 2.64 for the validation set. For the test set, the MAE was 2.33, the MSE was 9.72, the RMSE was 3.12, and the RSQUARED was 0.77. The RReliefF algorithm was also applied with a threshold of importance greater than 0.01. This algorithm considers the consistency of predictor values among neighbours with the same response values. The selected features and their importance scores were VF (0.1515), BFv (0.0142), and BFkv (0.0126), while DP had a negative importance score of -0.0092. The model's performance improved further, with an RMSE of 2.66, an MSE of 7.07, an RSQUARED of 0.86, and an MAE of 1.94 for the validation set. For the test set, the MAE was 1.95, the MSE was 7.39, the RMSE was 2.72, and the RSQUARED was 0.90. Table 2 presents the results of novel feature selection algorithms (NFSA). One NFSA algorithm was based on selecting variables with similar statistical characteristics, selecting TC, DP, VF, NPk, NPd, BFkv, and BFv. This algorithm achieved improved performance, with an RMSE of 1.74, an MSE of 3.01, an RSQUARED of 0.94, and an MAE of 1.14 for the validation set. For the test set, the MAE was 1.01, the MSE was 2.26, the RMSE was 1.50, and the RSQUARED was 0.95. The second NFSA algorithm focused on selecting variables with different statistical characteristics, selecting TC, DP, VF, NPk, NPmp, BFkv, and BFv. This algorithm achieved the best model performance, with an RMSE of 1.83, an MSE of 3.34, an RSQUARED of 0.94, and an MAE of 1.23 for the validation set. For the test set, the MAE was 0.99, the MSE was 2.14, the RMSE was 1.46, and the RSQUARED was 0.97. Based on the results from Table 2, it is evident that the novel feature selection algorithm with different statistical characteristics provided the best model performance, achieving the lowest RMSE for the validation set. This result emphasises and encourages researchers to develop models in this manner since it leads to better models in terms of accuracy and generalisation.

It is worth noting that further investigation and experimentation are necessary to validate the findings and potentially explore alternative modelling approaches or feature selection methods. The following study's limitations should also be acknowledged, such as the sample size, potential biases, and the context of the analysis. Future research could address these limitations to provide a more comprehensive understanding of the studied phenomena and potentially improve model performance by applying the novel feature selection algorithm for other scenarios like hybrid nanofluids and similar technologies.

Practical significance of the developed predictive model

By developing this model, it is possible to study and optimise nanofluids numerically before creating them. It enhances the ability to edit conventional fluids to fit any fluid description of our desire, especially the fluid's

Table 2 Novel feature selection algorithms (NFSAs)

Feature selection type	Brief description of the feature selection algorithm	Selected features and their importance	Model performance	
Novel feature selection algorithm is based on similar skewness and data resemblance	This selects variables that have close to or the same statistical characteristics	TC, DP, VF, NPk, NPd, BFkv, BFv	RMSE (Validation)	1.74
			MSE (Validation)	3.01
			RSQUARED (Validation)	0.94
			MAE (Validation)	1.14
			MAE (Test)	1.01
			MSE (Test)	2.26
			RMSE (Test)	1.50
			RSQUARED (Test)	0.95
Novel feature selection algorithm is based on different skewness and data resemblance. (The best)	This selects variables that have dissimilar. Statistical characteristics, differing values among neighbours with different response values	TC, DP, VF, NPk, NPmp, BFkv, BFv	RMSE (Validation)	1.83
			MSE (Validation)	3.34
			RSQUARED (Validation)	0.94
			MAE (Validation)	1.23
			MAE (Test)	0.99
			MSE (Test)	2.14
			RMSE (Test)	1.46
			RSQUARED (Test)	0.97

thermal properties. Also, it is to be noted that by editing the base fluids by adding nanoparticles, we can obtain numerous fluids (nanofluids as there are permutations of the features of the nanoparticles and base fluids) and adequately model their characteristics.

Conclusions

This research presents a novel approach for modelling single-material nanofluids, considering their constituents, the specific fluid characteristics, and the problems being addressed. The developed approach has demonstrated high accuracy in modelling nanofluids.

The significance of this study lies in its potential to advance our understanding of nanofluid behaviour by examining the individual and combined effects of variables on the thermophysical properties of nanofluids and providing researchers a road map on how to select features for nanofluid modelling so that we can have more general and accurate models. Furthermore, this methodological process for modelling as detailed in this study serves to suggest a process for researchers to apply when modelling nanofluids' thermophysical properties. By delving into these relationships, researchers can gain valuable insights into the underlying mechanisms governing nanofluid behaviour, leading to improved design and optimisation of nanofluid systems.

The ability to accurately model single material nanofluids opens up new possibilities for investigating and resolving the anomalous heat transfer enhancement observed in these fluids. Furthermore, it allows for the customisation of nanofluids to meet desired thermal properties, providing greater control over their application in various fields.

Overall, this research contributes to the growing body of knowledge on nanofluids, offering a promising avenue for further exploration and understanding of their thermophysical properties. The developed modelling approach sets the stage for future studies aimed at harnessing the full potential of nanofluids in enhancing heat transfer and achieving desired thermal characteristics. It is to be noted that the work is purely computational, and hence, researchers can look to validate these claims experimentally. Also applying these to hybrid nanofluid serves as a significant future work.

Appendix

Basic modelling and comparison step across various machine learning algorithms.

Model Type	RMSE (Validation)	MSE (Validation)	RSQUARED (Validation)	MAE (Validation)	MAE (Test)	MSE (Test)	RMSE (Test)	RSQUARED (Test)	Pre-set
Neural Network	1.707	2.912	0.947	1.196	0.792	1.547	1.244	0.964	Trilayered Neural Network
Gaussian Process Regression	1.871	3.501	0.936	1.299	1.006	2.185	1.478	0.949	Exponential GPR
Gaussian Process Regression	1.929	3.720	0.932	1.390	0.923	1.447	1.203	0.966	Squared Exponential GPR
Gaussian Process Regression	1.931	3.729	0.932	1.377	0.941	1.441	1.200	0.966	Matern 5/2 GPR
SVM	1.935	3.743	0.931	1.515	0.950	1.658	1.288	0.961	Quadratic SVM
Gaussian Process Regression	2.032	4.128	0.924	1.422	0.919	1.422	1.193	0.967	Rational Quadratic GPR
SVM	2.131	4.540	0.917	1.491	0.789	1.160	1.077	0.973	Cubic SVM
Neural Network	2.185	4.776	0.912	1.542	0.619	0.561	0.749	0.987	Narrow Neural Network
Neural Network	2.222	4.937	0.909	1.502	0.743	0.947	0.973	0.978	Bilayered Neural Network
Ensemble	2.359	5.567	0.898	1.672	1.165	2.062	1.436	0.952	Boosted Trees
Neural Network	2.529	6.395	0.883	1.641	0.655	0.790	0.889	0.982	Medium Neural Network
Stepwise Linear Regression	2.573	6.620	0.879	1.988	1.546	2.977	1.725	0.930	Stepwise Linear
Neural Network	2.590	6.707	0.877	1.632	0.797	1.761	1.327	0.959	Wide Neural Network
SVM	2.679	7.176	0.868	1.940	1.159	2.492	1.579	0.942	Medium Gaussian SVM
Ensemble	2.964	8.788	0.839	2.225	1.514	3.575	1.891	0.916	Bagged Trees
Tree	3.188	10.164	0.814	2.447	1.555	3.760	1.939	0.912	Fine Tree
Linear Regression	3.387	11.470	0.790	2.319	1.828	5.000	2.236	0.883	Interactions Linear
Linear Regression	3.530	12.459	0.771	2.889	2.045	6.297	2.509	0.853	Linear
Linear Regression	3.570	12.747	0.766	2.911	2.050	6.340	2.518	0.852	Robust Linear
SVM	3.712	13.777	0.747	2.949	2.001	7.062	2.657	0.835	Linear SVM
Tree	4.004	16.028	0.706	3.091	2.235	7.242	2.691	0.831	Medium Tree
SVM	4.394	19.305	0.646	3.193	2.393	11.682	3.418	0.727	Coarse Gaussian SVM
SVM	4.468	19.962	0.634	3.009	3.002	17.704	4.208	0.586	Fine Gaussian SVM

Model Type	RMSE (Validation)	MSE (Validation)	RSQUARED (Validation)	MAE (Validation)	MAE (Test)	MSE (Test)	RMSE (Test)	RSQUARED (Test)	Pre-set
Tree	5.164	26.672	0.511	4.082	3.579	19.630	4.431	0.541	Coarse Tree
Kernel	6.134	37.623	0.310	4.747	4.428	29.106	5.395	0.320	Least Squares Regression Kernel
Kernel	7.101	50.417	0.075	5.540	5.077	42.941	6.553	-0.003	SVM Kernel

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)
ENT	Percentage enhancement of nanofluid thermal conductivity (%)
GMDH	Group method of data handling
GPR	Gaussian process regressor
h	Mean value of observed heat transfer coefficient
MAE	Mean absolute error
ML	Machine learning
MRMR	Minimum redundancy maximum relevance
MSE	Mean-squared error
n	Number
NFSA	Novel feature selection algorithms
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
SSreg	Explained sum of squares
SStot	Total sum of squares
SVM	Support vector machine
TC	Nanofluid temperature (°C)
VF	Volume fraction (%)

Superscript

pred	Prediction

Subscript

i	Data point

Acknowledgements
The author wishes to thank Dr Jongrae Kim and Professor David Barton for their invaluable advice on the methods and presentation of the work. The author also wishes to thank the Tertiary Education Trust Fund (TET Fund) for providing funding for the studies and the University of Leeds for creating 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
All data sources were referenced in the manuscript body.

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: 27 June 2023 Accepted: 11 September 2023
Published online: 15 September 2023

References

Ahmadloo E, Azizi S (2016) Prediction of thermal conductivity of various nanofluids using artificial neural network. Int Commun Heat Mass Transf 74:69–75. https://doi.org/10.1016/j.icheatmasstransfer.2016.03.008
Breiman L (2001) Random forests. Mach Learn 45:5–32
Breiman L, Friedman J, Olshen R, Stone C (1984) Classification and regression trees. CRC Press, Boca Raton
Brownlee J (2020a) A gentle introduction to k-fold cross-validation. Retrieved May 5th 2022 from https://machinelearningmastery.com/k-fold-cross-validation/
Brownlee J (2020b) How to fix k-fold cross-validation for imbalanced classification. Retrieved May 27th 2022 from https://machinelearningmastery.com/cross-validation-for-imbalanced-classification/
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
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
Cong S, Zhou Y (2023) A review of convolutional neural network architectures and their optimizations. Artif Intell Rev 56(3):1905–1969. https://doi.org/10.1007/s10462-022-10213-5
Ewim DRE, Adelaja A, Onyiriuka E, Meyer J, Huan Z (2020) Modelling of heat transfer coefficients during condensation inside an enhanced inclined tube. J Therm Anal Calorim. https://doi.org/10.1007/s10973-020-09930-2
Ewim DRE, Okwu MO, Onyiriuka EJ, Abiodun AS, Abolarin SM, Kaood A (2021) A quick review of the applications of artificial neural networks (ANN) in the modelling of thermal systems
Friedman JH (2001) Greedy function approximation: a gradient boosting machine. Ann Stat. https://doi.org/10.1214/aos/1013203451
Gareth J, Daniela W, Trevor H, Robert T (2013) An introduction to statistical learning: with applications in R. Spinger, Berlin

Genzel M, Macdonald J, Marz M (2022) Solving inverse problems with deep neural networks-robustness included. IEEE Trans Pattern Anal Mach Intell. https://doi.org/10.1109/TPAMI.2022.3148324
Géron A (2022) Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow. O'Reilly Media, Inc, New York
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
Hastie T, Tibshirani R, Friedman JH, Friedman JH (2009) The elements of statistical learning: data mining, inference, and prediction, vol 2. Springer, Berlin
Hornik K, Stinchcombe M, White H (1989) Multilayer feedforward networks are universal approximators. Neural Netw 2(5):359–366. https://doi.org/10.1016/0893-6080(89)90020-8
Jiang C, Mi J, Laima S, Li H (2020) A novel algebraic stress model with machine-learning-assisted parameterization. Energies 13(1):258. https://doi.org/10.3390/en13010258
Kamsuwan C, Wang X, Piumsomboon P, Pratumwal Y, Otarawanna S, Chalermsinsuwan B (2023) Artificial neural network prediction models for nanofluid properties and their applications with heat exchanger design and rating simulation. Int J Therm Sci 184:107995. https://doi.org/10.1016/j.ijthermalsci.2022.107995
Kannaiyan S, Boobalan C, Nagarajan FC, Sivaraman S (2019) Modeling of thermal conductivity and density of alumina/silica in water hybrid nanocolloid by the application of Artificial Neural Networks. Chin J Chem Eng 27(3):726–736. https://doi.org/10.1016/j.cjche.2018.07.018
Komeilibirjandi A, Raffiee AH, Maleki A, Alhuyi Nazari M, Safdari Shadloo M (2020) Thermal conductivity prediction of nanofluids containing CuO nanoparticles by using correlation and artificial neural network. J Therm Anal Calorim 139:2679–2689. https://doi.org/10.1007/s10973-019-08838-w
Kurani A, Doshi P, Vakharia A, Shah M (2023) A comprehensive comparative study of artificial neural network (ANN) and support vector machines (SVM) on stock forecasting. Ann Data Sci 10(1):183–208. https://doi.org/10.1007/s40745-021-00344-x
Maronna RA, Martin RD, Yohai VJ, Salibián-Barrera M (2019) Robust statistics: theory and methods (with R). Wiley, New York
MathWorks (2022) Statistics and machine learning toolbox: documentation (R2022a).
Meng M, Zhong R, Wei Z (2020) Prediction of methane adsorption in shale: Classical models and machine learning based models. Fuel 278:118358. https://doi.org/10.1016/j.fuel.2020.118358
Mijwil, M. M. (2018). Artificial Neural Networks Advantages and Disadvantages. https://www.linkedin.com/pulse/artificial-neural-networks-advantages-disadvantages-maad-m-mijwel/
Mohammed A, Kora R (2023) A comprehensive review on ensemble deep learning: Opportunities and challenges. J King Saud Univ Comput Inf Sci. https://doi.org/10.1016/j.jksuci.2023.01.014
Moran MJ, Shapiro HN, Boettner DD, Bailey MB (2010) Fundamentals of engineering thermodynamics. Wiley, New York
Murshed S, Leong K, Yang C (2005) Enhanced thermal conductivity of TiO2—water based nanofluids. J Int J Therm Sci 44(4):367–373. https://doi.org/10.1016/j.ijthermalsci.2004.12.005
Onyiriuka EJ (2023a) Predicting the accuracy of nanofluid heat transfer coefficient's computational fluid dynamics simulations using neural networks. Heat Transf
Onyiriuka EJ (2023b) Single phase nanofluid thermal conductivity and viscosity prediction using neural networks and its application in a heated pipe of a circular cross section. Heat Transf
Osborne J (2010) Improving your data transformations: Applying the Box-Cox transformation. Pract Assess Res Eval 15(1):12. https://doi.org/10.7275/qbpc-gk17
Patel H (2021) What is feature engineering—importance, tools and techniques for machine learning. Medium. Retrieved 15th July from https://towardsdatascience.com/what-is-feature-engineering-importance-tools-and-techniques-for-machine-learning-2080b0269f10
Patel HE, Sundararajan T, Das SK (2010) An experimental investigation into the thermal conductivity enhancement in oxide and metallic nanofluids. J Nanopart Res 12(3):1015–1031. https://doi.org/10.1007/s11051-009-9658-2
Peng Y, Parsian A, Khodadadi H, Akbari M, Ghani K, Goodarzi M, Bach Q-V (2020) Develop optimal network topology of artificial neural network (AONN) to predict the hybrid nanofluids thermal conductivity according to the empirical data of Al2O3–Cu nanoparticles dispersed in ethylene glycol. Physica A 549:124015. https://doi.org/10.1016/j.physa.2019.124015
Ramezanizadeh M, Alhuyi Nazari M, Ahmadi MH, Lorenzini G, Pop I (2019) A review on the applications of intelligence methods in predicting thermal conductivity of nanofluids. J Therm Anal Calorim 138(1):827–843. https://doi.org/10.1007/s10973-019-08154-3
Razavi R, Sabaghmoghadam A, Bemani A, Baghban A, Chau K-W, Salwana E (2019) Application of ANFIS and LSSVM strategies for estimating thermal conductivity enhancement of metal and metal oxide based nanofluids. Eng Appl Comput Fluid Mech 13(1):560–578. https://doi.org/10.1080/19942060.2019.1620130
Rish I (2001) An empirical study of the naive Bayes classifier. In: IJCAI 2001 workshop on empirical methods in artificial intelligence.
Sharma P, Ramesh K, Parameshwaran R, Deshmukh SS (2022) Thermal conductivity prediction of titania-water nanofluid: A case study using different machine learning algorithms. Case Stud Therm Eng 30:101658. https://doi.org/10.1016/j.csite.2021.101658
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/10.1063/5.0099553
Vapnik V (1999) The nature of statistical learning theory. Springer, Berlin
Witten IH, Frank E, Hall MA, Pal CJ (2016) Data mining: practical machine learning tools and techniques. Morgan Kaufmann, New York
Xie H, Wang J, Xi T, Liu Y, Ai F, Wu Q (2002) Thermal conductivity enhancement of suspensions containing nanosized alumina particles. J Appl Phys 91(7):4568–4572. https://doi.org/10.1063/1.1454184
Yashawantha KM, Vinod AV (2021) ANN modelling and experimental investigation on effective thermal conductivity of ethylene glycol: water nanofluids. J Therm Anal Calorim 145(2):609–630. https://doi.org/10.1007/s10973-020-09756-y
Zhu G, Wen T, Zhang D (2021) Machine learning based approach for the prediction of flow boiling/condensation heat transfer performance in mini channels with serrated fins. Int J Heat Mass Transf 166:120783. https://doi.org/10.1016/j.ijheatmasstransfer.2020.120783

Publisher's Note