scientific reports

OPEN

Development of novel computational models based on artificial intelligence technique to predict liquids mixtures separation via vacuum membrane distillation

Yanfen Wei

The fundamental objective of this paper is to use Machine Learning (ML) methods for building models on temperature (T) prediction using input features r and z for a membrane separation process. A hybrid model was developed based on computational fluid dynamics (CFD) to simulate the separation process and integrate the results into machine learning models. The CFD simulations were performed to estimate temperature distribution in a vacuum membrane distillation (VMD) process for separation of liquid mixtures. The evaluated ML models include Support Vector Machine (SVM), Elastic Net Regression (ENR), Extremely Randomized Trees (ERT), and Bayesian Ridge Regression (BRR). Performance was improved using Differential Evolution (DE) for hyper-parameter tuning, and model validation was performed using Monte Carlo Cross-Validation. The results clearly indicated the models' effectiveness in temperature prediction, with SVM outperforming other models in terms of accuracy. The SVM model had a mean R^2 value of 0.9969 and a standard deviation of 0.0001, indicating a strong and consistent fit to the membrane data. Furthermore, it exhibited the lowest mean squared error, mean absolute error, and mean absolute percentage error, signifying superior predictive accuracy and reliability. These outcomes highlight the importance of selecting a suitable model and optimizing hyperparameters to guarantee accurate predictions in ML tasks. It demonstrates that using SVM, optimized with DE improves accuracy and consistency for this specific predictive task in membrane separation context.

Keywords Separation, CFD, Machine learning, Membrane distillation, Support vector machine

Abbreviations

ANFIS	Adaptive neuro-fuzzy inference system
BRR	Bayesian ridge regression
CFD	Computational fluid dynamics
CR	Crossover rate
DE	Differential evolution
ENR	Elastic Net regression
ERT	Extremely randomized trees
MD	Membrane distillation
mem	Membrane
ML	Machine learning
MCCV	Monte Carlo cross-validation
MSE	Mean squared error
MAE	Mean absolute error
MAPE	Mean absolute percentage error
RMSE	Root mean squared error
SVM	Support vector machine
VMD	Vacuum membrane distillation

Department of Information Management, School of Big Data and Artificial Intelligence, Guangxi University of Finance and Economics, Nanning 530003, China. email: wyf_1999_whu@126.com

Notations

A	Species A
C	Regularization parameter in SVM
C_A	Concentration of species (mol/m^3)
C_p	Specific heat capacity (J/kg K)
D_A	Diffusion coefficient of compound A (m^2/s)
F	Scaling factor
K	Thermal conductivity (W/m K)
N_A	Mass transfer flux of compound A (mol/m^2 s)
r	Radial coordinate (m)
r_1	Inner radius of membrane (m)
R^2	Coefficient of determination
T	Temperature (K)
t	Tube side of membrane
u	Average fluid velocity in tube side (m/s)
V	Velocity of fluid (m/s)
w	Weight parameter
y_i	Observed response
z	Axial coordinate (m)
β	Regression coefficients
α	Hyper-parameter controlling the mix between L1 and L2 regularization
λ	Regularization parameter
μ	Mean vector of the posterior distribution
Σ	Covariance matrix of the posterior distribution
ξ	Slack variable in SVM
$\in$	Insensitivity parameter in SVM

Separation of liquid solutions can be conducted via membrane systems where different methods have been developed so far such as ultrafiltration, nanofiltration, reverse osmosis, etc. The size of membrane pores and the separation mechanism are different for each membrane separation technique. An important class of membrane separation technique is membrane contactor where these units are used for contacting two phases such as liquid-liquid contact to separation one compound or conduct chemical reaction[1–3]. Membrane contactors can offer various operations due to their unique characteristics in process engineering. Ind fact, various unit operations can be done via membrane contactors such as chemical reactions in membrane contactors where it can be driven in continuous mode of operation[4,5].

Combination of membrane contactors with other unit operations can be considered as a novel approach for developing hybrid processing. Such hybrid processes have been developed already such as membrane distillation (MD) where distillation is conducted in continuous flow via membrane contactors[6–8]. For direct-contact membrane distillation, the hot and cold fluids are brought into contact in the membrane contactor and separation occurs in the process.

Other type of MD is vacuum membrane distillation (VMD) which relies on creating pressure difference between the feed and permeate side for separation of compounds based on volatility[9]. VMD has privilege that it does not need additional cold phase to make driving force between two phases[10,11]. It can also operate in the ambient condition for separation. The process can be developed and optimized by computational techniques such as computational fluid dynamics (CFD) which is used for numerical solution of transport phenomena in membrane process[12,13]. Temperature distribution and concentration of solute in the feed can be obtained by this method. Due to the complexity of CFD, other approaches such as machine learning models can be employed for simulation of this process.

MD has been previously simulated using ANFIS (Adaptive Neuro-Fuzzy Inference System) method by using some CFD simulations dataset and the results were promising in terms of accuracy[14]. Also, concentration distribution of solute in VMD process has been obtained via combination of CFD and ML models[15]. However, wider analysis is required to explore more machine learning models for analysis of VMD process. Indeed, there is a research gap on development of combined hybrid CFD and machine learning models for VMD process which provides holistic view of the process. This is addressed in this study by developing hybrid CFD-machine learning models for VMD process to predict temperature distribution in the process.

This study explores the effectiveness of different machine learning models in predicting temperature (T) using the input features r and z which are the membrane system's coordinates. The selected models include Bayesian Ridge Regression (BRR), Extremely Randomized Trees (ERT), Elastic Net Regression (ENR), and Support Vector Machine (SVM). The selected models encompass a blend of linear and non-linear methodologies, enabling a thorough assessment of their predictive performance.

BRR amalgamates Bayesian principles with ridge regression, estimating regression coefficients by integrating prior information and introducing a regularization term to mitigate overfitting. This strategy improves the model's generalization by harmonizing the model's fit and complexity via regularization. ERT is an ensemble learning method that constructs multiple unpruned decision trees using random cut-points and the entire training dataset. This technique reduces variance and computational complexity by averaging the predictions from these randomized trees, improving overall prediction accuracy. ENR combines L1 (Lasso) and L2 (Ridge) regularization to handle multicollinearity and promote sparsity. The SVM is a resilient model that effectively identifies the optimal hyperplane in a higher-dimensional space to separate data points efficiently. The primary goal is to maximize the margin between the support vectors and the hyperplane, resulting in minimal training error and enhanced generalization. This methodology is beneficial for both classification and regression assignments. To ensure robust predictions, the study utilized hyper-parameter optimization through Differential Evolution (DE) and model validation via Monte Carlo Cross-Validation (MCCV). DE is an evolutionary optimization technique adept at handling multidimensional data, while MCCV involves multiple random splits of the dataset into training and testing subsets to validate model performance comprehensively. Performance metrics such as R^2 score, Mean Squared Error (MSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE) were calculated to assess each model.

Process data

The dataset employed in this investigation are more than 13,000 data instances, each characterized by three variables: r, z, and T (temperature of fluid). The initial two coordinates serve as inputs, while T represents the sole output of this analysis. r and z represent the radial and axial coordinates in the membrane system, respectively where the temperature of solution is calculated via CFD simulations.

For data collection, a hollow-fiber membrane contactor for VMD was simulated by utilizing *COMSOL* package and numerical solution of mass and energy equations. *UMFPACK* was used as the direct solver in numerical calculations[15]. The meshed geometry of the model for the feed and membrane sides is illustrated in Fig. 1. As seen, the system is divided into two parts, i.e., feed and membrane. The equations of convection-diffusion for mas transfer and convection-conduction for the heat transfer were solved for the geometry using finite element method. A constant concentration and temperature boundary condition was assumed for the inlet of feed channel, whereas convective flow is assumed for the outlet of feed channel. Separation of an organic compound from aqueous solution was considered in this study. The flow regime was assumed to be laminar due to the low flow rate and small channel dimension in the membrane system[14]. Once the simulations have been carried out, the temperature distribution entire the feed side was collected for ML modeling. So, the temperature as a function of coordinates was obtained for machine learning modeling in this study. The used equations in this work for CFD simulations are listed in Table 1. It should be noted that the CFD methodology used in this study has been already validated through comparing with experimental data as reported in[16,17].

The pair plot depicted in Fig. 2 offers an extensive perspective on the interconnections among the variables. The histograms on the diagonal represent the distribution of each variable, while the scatter plots off the diagonal illustrate the relationships between each pair of variables. The scatter plots indicate a significant negative correlation between z and T, implying that as z increases, T generally decreases. The association between r and T appears to be less pronounced, indicating a relatively weaker relationship.

The box plots in Fig. 3 illustrate the distribution and variability of each variable. For r and z, the data points are tightly clustered around the median, indicating low variability. T, the output variable, also shows a symmetric distribution with a median around 318 K and a slight spread, indicating moderate variability.

Methodology

The methodology for predicting T based on the input variables r and z involves several key steps: dataset preparation, outlier detection, model selection, hyper-parameter optimization, model validation, and performance evaluation[14]. As the dataset is complete with no missing values, no preprocessing for missing data was necessary. However, to ensure data quality, outliers were identified and handled using the Elliptic Envelope method (here less than 0.5% of data detected as outliers). The data also normalized using Min-Max scaler. Four ML models were chosen: BRR, ERT, ENR, and SVM. These models were selected to provide a mix of linear and non-linear approaches, allowing a comprehensive evaluation of their predictive capabilities. Hyper-parameters for each model were optimized using DE. Model validation was performed using MCCV. MCCV involves randomly splitting the dataset into training and testing subsets multiple times, training the model on the training subset, and evaluating it on the testing subset. This process was repeated to ensure robust validation of model performance, with mean and standard deviation of the performance metrics being reported. This method guarantees that a single train-test split does not bias the validation results and offers a more complete evaluation of the generalizability of the model[18]. Here the number of repeats is set to 100 and the train and test subsets are 80% and 20% of the entire dataset. Using R^2 score, MSE, MAE, and MAPE, each model's performance was assessed. Each metric was computed for the models across multiple MCCV iterations to obtain the mean and standard deviation, providing a comprehensive view of the model performance. This systematic approach ensured a thorough evaluation of the selected models, identifying the most accurate and reliable model for predicting temperature based on the input variables r and z. More detailed descriptions of the building blocks that are mentioned here are included in the following subsections. Step-by-step implementation of the proposed approach is shown in Fig. 4. Machine learning algorithms and optimizers implemented using *Python* programming language.

Elliptic envelope method for outlier detection

The Elliptic Envelope method is a statistical technique for detecting outliers by assuming that the data follows a multivariate Gaussian distribution. This method constructs an elliptical boundary that encompasses the majority of the data points, effectively capturing the central tendency and dispersion of the dataset. By fitting an ellipse to the data, the method identifies points that lie outside this boundary as outliers[19]. The process begins by estimating the mean and covariance of the dataset. The Elliptic Envelope then uses these estimates to create an ellipse that encompasses the central data points, defined by a chosen confidence level. Mathematically, it aims to minimize the volume of the ellipse while covering a specified proportion of the data, typically set to 95% or 99%. Data points lying outside this ellipse are flagged as outliers.

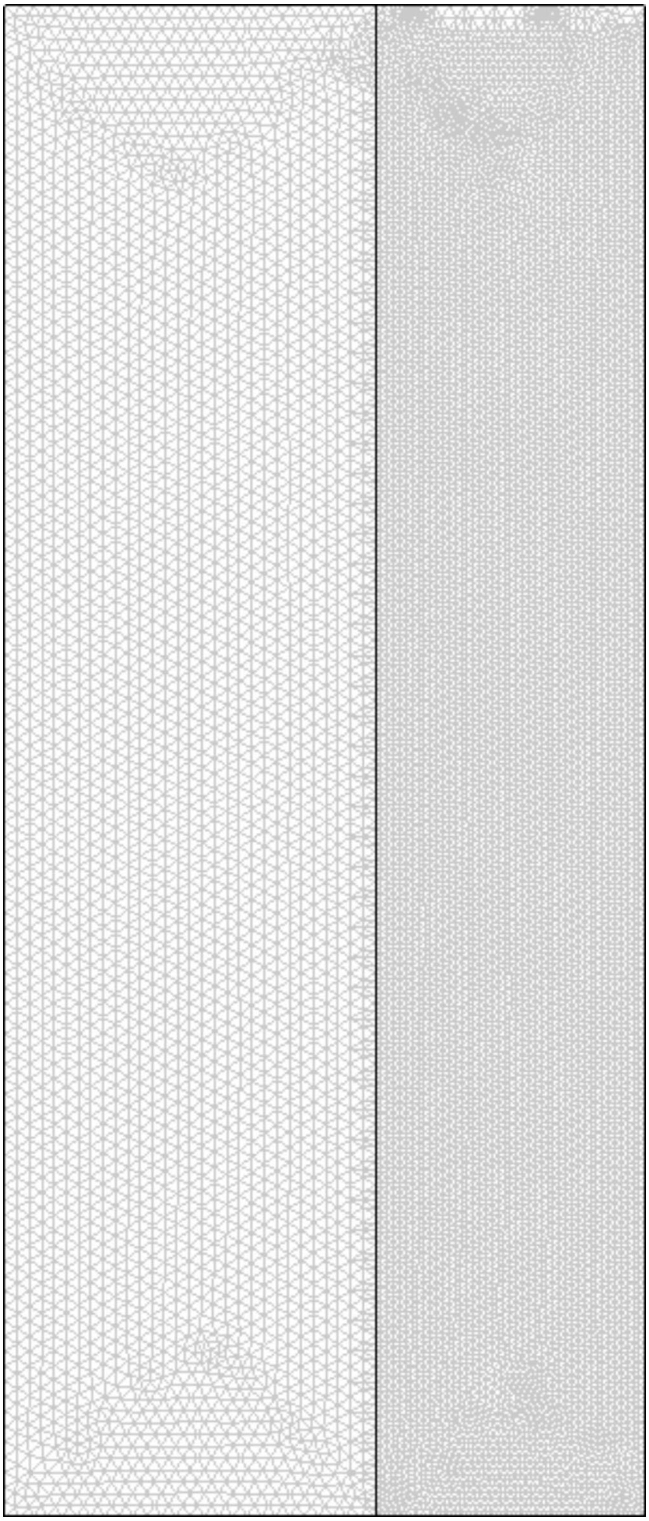

Fig. 1. Meshed geometry of VMD process for CFD simulations.

Differential evolution

DE is recognized as a meta-heuristic evolutionary optimization approach aimed at iteratively enhancing the quality of potential solutions to optimize a given problem. This optimizer can effectively handle multidimensional real-valued data, even when the function to be optimized lacks differentiability. Moreover, it can address issues that are noisy, discontinuous, or dynamic. DE operates by utilizing a population of feasible solutions, combining them through simple mathematical operations to identify the solution that is most optimal for the optimization problem. The algorithm encodes the variables of the problem as a vector of real numbers. The population consists

Domain	Equation	No.
Mass transfer for solute		
Continuity	$N_A = -D_A \nabla C_A + V_z C_A$	1
Tube	$D_{A,t} \left[\frac{1}{r}\frac{\partial}{\partial r}\left(r \frac{\partial C_{A,t}}{\partial r} \right) + \frac{\partial^2 C_{A,t}}{\partial z^2} \right] = V_{z,t} \frac{\partial C_{A,t}}{\partial z}$	2
Membrane	$D_{A,mem} \left[\frac{\partial^2 C_{A,mem}}{\partial r^2} + \frac{\partial^2 C_{A,mem}}{\partial z^2} \right] = 0$	3
Heat transfer		
Tube	$K_t \left[\frac{1}{r}\frac{\partial}{\partial r}\left(r \frac{\partial T}{\partial r} \right) + \frac{\partial^2 T}{\partial z^2} \right] = \rho_t C_{p,t} V_{z,t} \frac{\partial \mathrm{T}}{\partial z}$	4
Membrane	$K_{mem} \left[\frac{1}{r}\frac{\partial}{\partial r}\left(r \frac{\partial T_{mem}}{\partial r} \right) + \frac{\partial^2 T_{mem}}{\partial z^2} \right] = 0$	5
Momentum transfer		
Tube	$V_{z,t} = 2u \left[1 - \left(\frac{r}{r_1} \right)^2 \right]$	6

Table 1. Governing equations for membrane process used in this study[14,16,17].

of vectors of length *n*, corresponding to the parameters in the optimization problem[20,21]. A vector is represented by $x_{g,p}$, where *p* indicates its index within the population, and *g* denotes its generation. The components of this vector, $x_{g,p,m}$ are bounded within intervals defined by $x_{\min,m}$ and $x_{\max,m}$. The DE algorithm involves four stages including initialization, mutation, recombination, and selection[22]. The algorithm iterates through these stages until a stopping criterion is met, which can be based on the number of generations, time elapsed, or the degree of optimization achieved[23,24].

In this study, DE was utilized to optimize the hyperparameters of the machine learning models (SVM, ENR, ERT, BRR). The fitness function which should be maximized is set to R^2 score. The algorithm parameters were set as follows:

- Population size: 50.
- Scaling factor (F): 0.8.
- Crossover rate (CR): 0.9.
- Number of generations: 100.

The DE algorithm iteratively adjusted the hyperparameters to minimize the objective function, which in this case was the sum of the Root Mean Squared Error (RMSE) and MAE values, considering their standard deviations.

Bayesian ridge regression

Bayesian statistics and ridge regression form a robust regression data analysis model. This method models the relationship between independent input variables and a continuous dependent response in a flexible and robust manner[25]. It calculates the coefficients of a linear model through considering prior knowledge about the data using a prior distribution. The posterior distribution is obtained by combining this distribution with the likelihood function. It is used to inform the estimation of coefficients and predictions of data. The technique incorporates a penalty term to reduce overfitting, thereby enhancing generalization. The regression coefficients conform to a normal distribution with a mean of zero and a variance that is determined by the hyper-parameter alpha. The likelihood function is modeled as a normal distribution, with a linear regression determining the mean and an additional hyperparameter, lambda, controlling the variance. The main objective is to infer the most likely values for the regression coefficients beta based on the given data and prior information. Following equation is the posterior distribution of *β*[26]:

$$p(\beta \mid X, y, \alpha, \lambda) = \mathrm{N}(\beta \mid \mu, \Sigma) \tag{7}$$

In this equation, *μ* represents the mean vector and *Σ* indicates the covariance matrix of the posterior distribution. The Bayesian formula is utilized to determine these parameters[27]:

$$\mu = (\lambda \cdot X' X^{\cdot} + \alpha \cdot I)^{-1} \cdot X' y^{\cdot} \tag{8}$$

$$\Sigma = (\lambda \cdot X' X^{\cdot} + \alpha \cdot I)^{-1} \tag{9}$$

where X'_y stands for the transposed input matrix multiplied by the output variable, $X'X$ indicates the transpose of the input variable matrix multiplied by itself, and I denotes the identity matrix.

Extremely randomized trees

ERT is an ensemble learning method introduced by Geurts et al.[28] in 2006. Tree-based regressors consist of hierarchical rule sets that predict numerical output values. By averaging the randomized predictions from

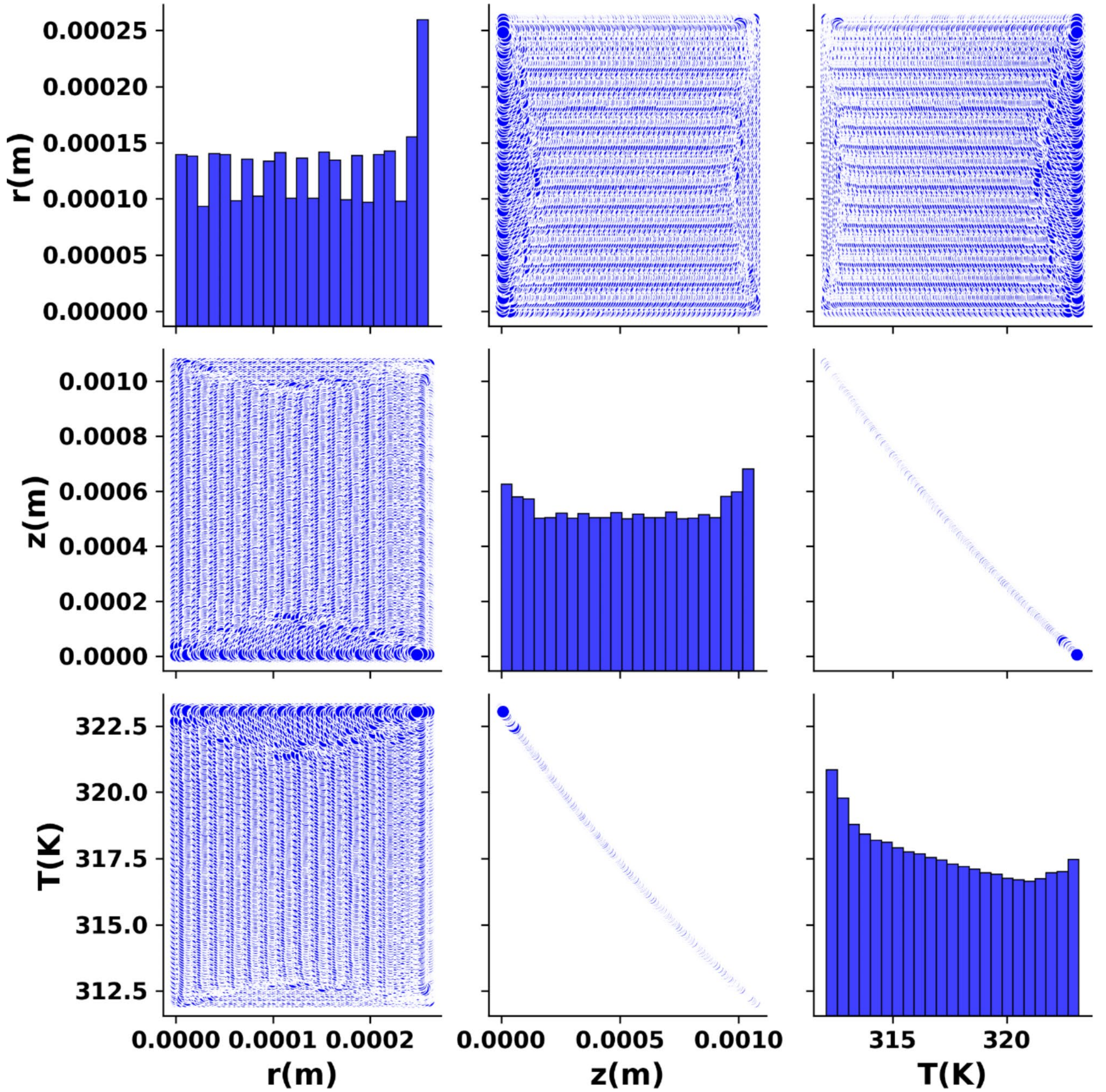

Fig. 2. Pair plot of the dataset obtained from VMD process.

multiple decision trees, Extra-Trees enhance prediction accuracy while significantly reducing computational complexity[29]. The extra-trees method relies on the concept of the bias-variance trade-off. The utilization of explicit randomization of cut-points and characteristics, along with ensemble averaging, efficiently diminishes variation to a greater extent compared to less assertive randomization tactics observed in alternative algorithms. To minimize bias, the entire original training set is used instead of bootstrap samples. The computational complexity of growing the trees, assuming balanced trees, is on the order of $\log N$ relative to the size of the training sample, similar to other tree-growing algorithms. Moreover, owing to the straightforward nature of the node splitting procedure, the constant factor is anticipated to be significantly lower compared to earlier ensemble methods that optimize cut-points locally.

Elastic Net

Elastic Net is a potent technique that synergistically integrates the advantages of L1 (Lasso) and L2 (Ridge) regularization techniques. L1 regularization encourages sparsity by shrinking some coefficients to exactly zero, thereby performing variable selection and enhancing interpretability. Concurrently, L2 regularization mitigates multicollinearity issues and stabilizes the coefficient estimates by shrinking them towards zero without eliminating any variables completely[30,31]. The Elastic Net model introduces two key hyperparameters: α and λ.

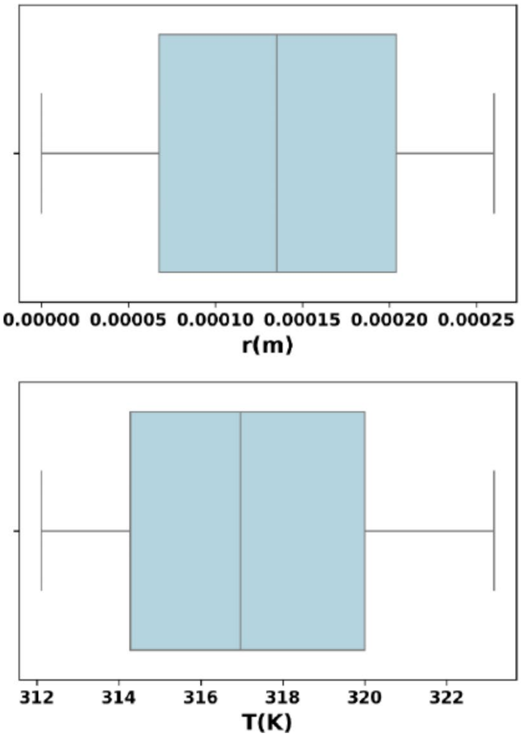

Fig. 3. Box plot of variables extracted from VMD simulation.

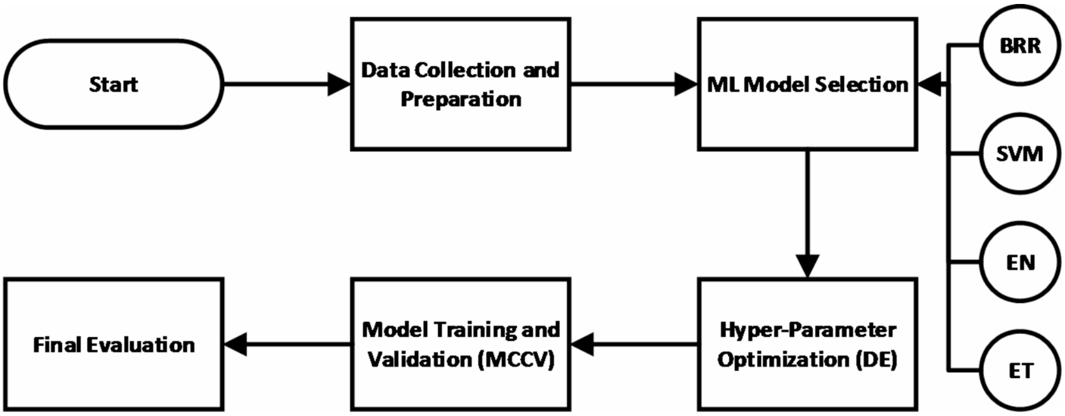
Start
Data Collection and Preparation
ML Model Selection
BRR
SVM
EN
ET
Hyper-Parameter Optimization (DE)
Model Training and Validation (MCCV)
Final Evaluation

Fig. 4. General methodology designed for this work.

The α parameter controls the mix between L1 and L2 regularization. When $\alpha = 1$, the model reduces to Lasso regression, and when $\alpha = 0$, it becomes Ridge regression. The λ parameter determines the overall strength of the regularization applied to the model, with higher values leading to more regularization. The objective function for Elastic Net combines the penalties from both Lasso and Ridge[32]:

$$\text{minimize}\left(\frac{1}{2n}\sum_{i=1}^{n}\left(y_i - X_i \cdot \beta\right)^2 + \lambda\left(\alpha\sum_{j=1}^{p}|\beta_j| + \frac{1-\alpha}{2}\sum_{j=1}^{p}\beta_j^2\right)\right) \tag{10}$$

In this equation, n denotes the observations, y_i stands for the observed output, and X_i represents the vector of predictors for the i-th observation. The vector β represents the coefficients to be determined from dataset. The term α balances L1 and L2 penalties, while λ denotes the regularization parameter. Finding the optimum values for α and λ can be achieved through grid search with cross-validation. This method systematically evaluates different combinations of α and λ to identify the best-performing model configuration. This approach ensures that the model is well-tuned and performs optimally on the validation set, thereby enhancing its generalizability to new data. Combining the flexibility of Lasso and the strength of Ridge gives Elastic Net Regularization a well-balanced method for linear regression. For datasets with high-dimensional characteristics or multicollinearity especially it is helpful.

Support vector machine

SVM is a popular ML model, particularly adept at fitting data by identifying a line or hyperplane in a higher-dimensional space. The hyperplane is defined by a set of data points known as support vectors, which are the most closely correlated with it. The goal of SVM regression is to find the hyperplane that maximizes the margin between the support vectors and the hyperplane while reducing the training error[33–35]. In SVM regression, we aim to find a function $f\left(x\right) = w\cdot x + b$ that deviates from the actual target values y_i by no more than $\in$, ensuring the function is as flat as possible. The objective is to minimize[36]:

$$\frac{1}{2}|w|^2$$

Considering:

$$y_i - \left(w\cdot x_i + b\right) \le$$

$$\left(w\cdot x_i + b\right) - y_i \le$$

To accommodate errors, slack variables ξ_i and ξ_i^* are applied in the model, leading to the modified objective[36]:

$$\frac{1}{2}|w|^2 + C\sum_{i=1}^{n}\left(\xi_i + \xi_i^*\right)$$

According to:

$$y_i - \left(w\cdot\ x_i + b\right) \le \in +\xi_i$$

$$\left(w\cdot x_i + b\right) - y_i \le \in +\xi_i^*$$

$$\xi_{\ i}, \xi_{\ i}^{\ *} \ge\ 0$$

Here, *C* is a regularization parameter balancing flatness and tolerance for deviations. In SVM regression, the mapping between the input and output spaces is determined by the kernel function. Common kernel functions comprise the radial basis function (RBF), linear, polynomial, and sigmoid. Choosing the kernel function is dependent on the unique problem, the quantity of features, and the noise level within the data[37].

Results and discussion

The performance of four distinct ML models is evaluated for predicting T based on the input features r and z in the domain of membrane contactor. The models tested were BRR, ERT, ENR, and SVM. DE was used for hyper-parameter optimization, and MCCV was employed for model validation. The evaluation metrics taken into account included the mean R^2 score, MSE, MAE, and MAPE, along with their corresponding standard deviations. Table 2 offers a succinct overview of each model's performance across various metrics.

The results indicate that the SVM performed the best among the other models for fitting the temperature dataset in the MD process. Specifically, the SVM model achieved the highest mean R^2 score (0.9969) with the lowest standard deviation (0.0001), indicating a very strong and consistent fit to the data. Additionally, it had the lowest mean MSE (0.03313), MAE (0.15803), and MAPE (4.9774%), demonstrating its superior predictive accuracy and robustness.

The ENR model also performed very well, with a mean R^2 score of 0.9912 and significantly lower errors compared to the BRR and ERT models. The ENR model's mean MSE (0.09349), MAE (0.24718), and MAPE (7.7591%) were higher than those of the SVM but still substantially better than BRR and ET.

The ERT model showed moderate performance, with a mean R^2 score of 0.9526 and relatively lower errors compared to BRR but higher than both ENR and SVM. The BRR model exhibited the lowest performance among the models tested, with the highest errors and lowest R^2 score.

Figure 5 displays a comparison between the actual values obtained through experimentation (validated CFD simulations) and the values predicted by utilizing all four ML models. Furthermore, Fig. 6 illustrates the distribution of R^2 scores obtained through MCCV method.

Model	MCCV Mean R^2 score	R^2 score std dev	MCCV mean MSE	MSE std dev	MCCV mean MAE	MAE std dev	MCCV mean MAPE	MAPE std dev
BRR	0.9412	0.0020	0.62530	0.02579	0.67574	0.01534	21.2802%	0.48297%
ERT	0.9526	0.0010	0.51089	0.01624	0.61960	0.01144	19.5139%	0.35950%
ENR	0.9912	0.0002	0.09349	0.00350	0.24718	0.00424	7.7591%	0.13131%
SVM	0.9969	0.0001	0.03313	0.00057	0.15803	0.00178	4.9774%	0.05599%

Table 2. Monte Carlo cross-validation performance metrics (Test data).

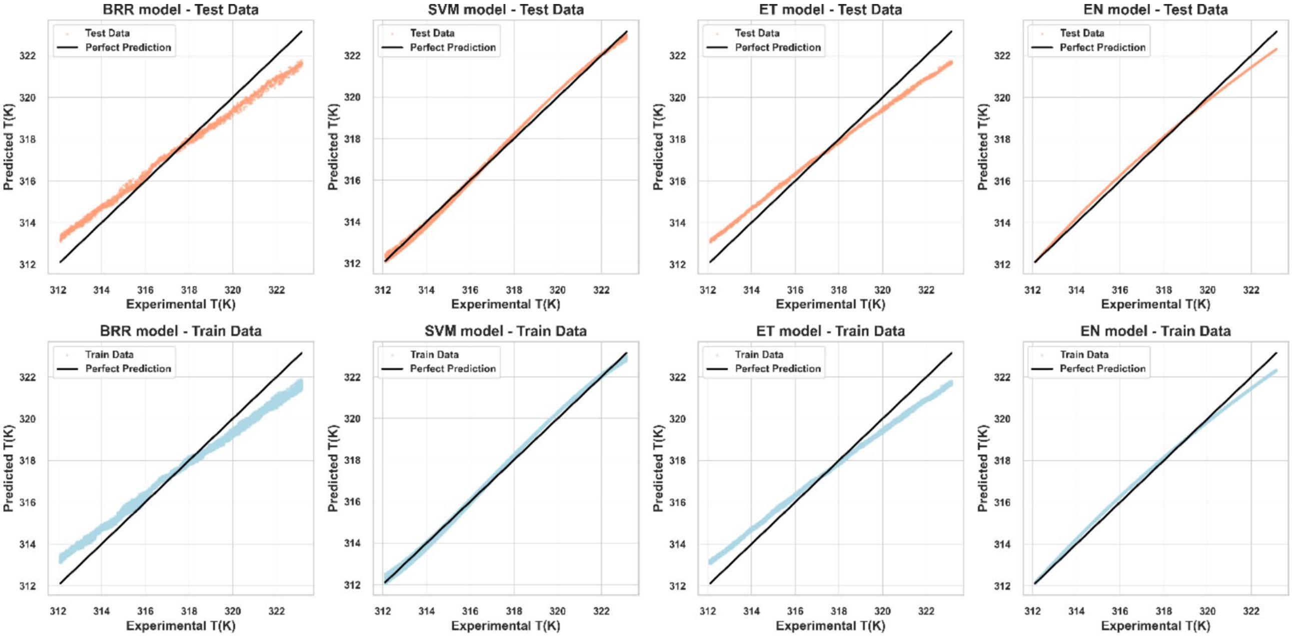

Fig. 5. CFD (*Experimental*) versus estimated values using BRR, SVM, ERT, and ENR models.

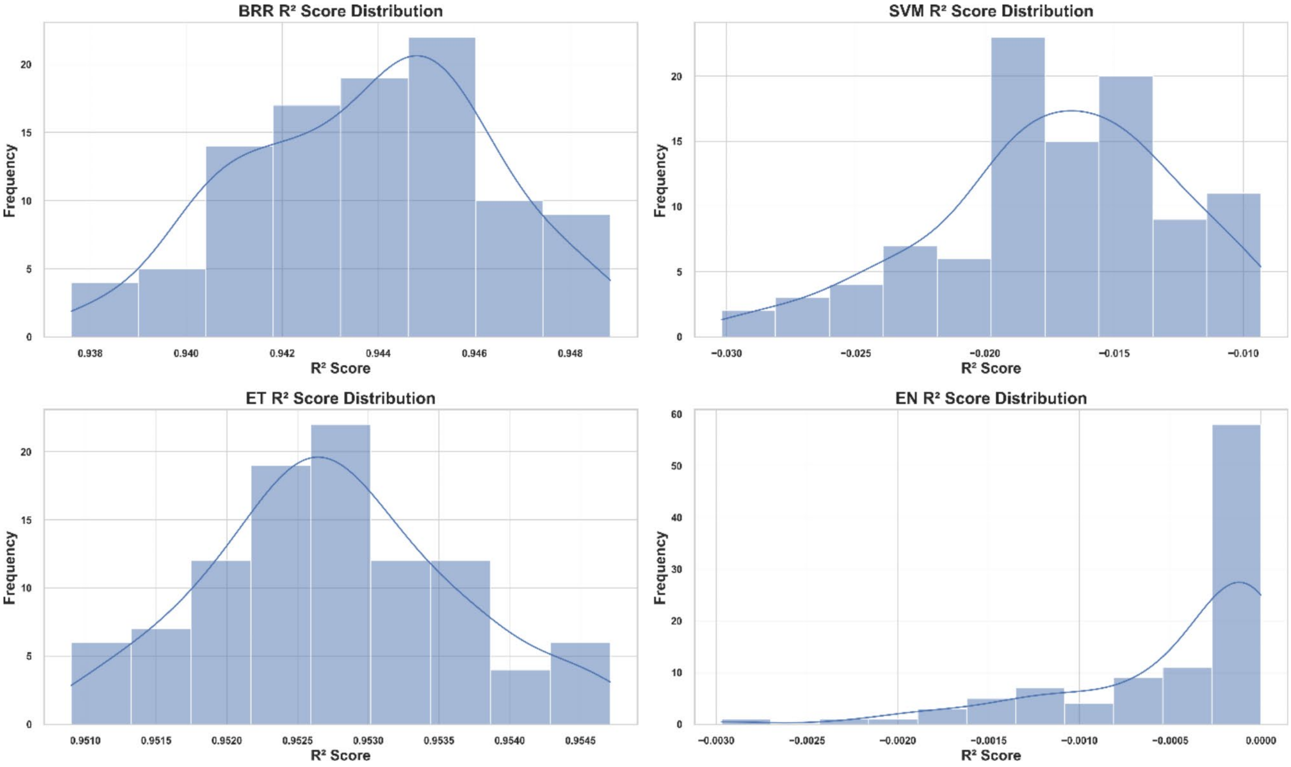

Fig. 6. Distribution of R^2 scores from Monte Carlo cross-validation (BRR, SVM, ERT, and ENR models).

Overall, the SVM model has superior performance in predicting temperature using the input features r and z. It regularly obtained the greatest R^2 scores and the lowest error metrics, demonstrating exceptional predictive accuracy and reliability. The ENR model also demonstrated strong performance and may be regarded as a viable alternative if SVM is deemed less suitable for specific applications due to factors such as interpretability or other reasons. The ERT and BRR models, albeit less precise, can nonetheless be useful in situations when their particular strengths or computational benefits are applicable.

The contour plot of T is given in Fig. 7, using SVM as the most suitable model. Additionally, Fig. 8 presents a 3D depiction of T(K) based on coordinates. Finally, the partial dependencies between the inputs and the temperature T(K) are illustrated in Fig. 9. It is seen that the temperature is decreased along the membrane process which is due to the evaporation of the feed solution which in turn decreases the system's temperature. In radial direction, a parabolic change for temperature can be observed which could be attributed to the convective term which is dominant in the membrane process in axial direction. This term is mainly due to the fluid flow in axial direction and effect of viscous forces in the solution which causes a parabolic velocity profile in the feed channel of membrane contactor[14].

While the DE algorithm effectively optimized the machine learning models' hyperparameters, there are potential drawbacks. Overfitting is a concern where models excel on training data but perform poorly on new data. This risk is mitigated by MCCV in this study. DE's computational costs can be high, especially with large populations and many generations, leading to longer processing times. Efficient computing resources and parallel processing helped to address this. Future study can be performed on development of hybrid optimization techniques to minimize the deviation of ML estimated parameters from the reference data, while avoiding biased models.

Conclusion

This study evaluated the performance of four machine learning models—BRR, ERT, ENR, and SVM—for predicting temperature based on the input variables. The models are trained via dataset obtained from CFD simulations of VMD process for separating liquid mixtures. Using hyper-parameter optimization through DE and model validation via MCCV, each model's predictive capabilities was assessed. The findings indicated that the SVM model exhibited superior performance compared to the other models in all performance metrics. It attained the highest average R^2 score of 0.9969 and the lowest average values for MSE, MAE, and MAPE. The findings emphasize the superior predictive accuracy and robustness of SVM, establishing it as the most dependable model for this particular predictive task. The ENR model exhibited robust performance, displaying noteworthy accuracy and lower error rates in comparison to BRR and ERT. The ERT model demonstrated a moderate level of performance, whereas the BRR model displayed the lowest level of performance among the models that were tested. In summary, this study emphasizes the significance of choosing suitable machine learning models and fine-tuning their hyper-parameters to achieve precise predictions. Subsequent investigations could focus on refining these models and testing their effectiveness on different datasets to confirm their ability to be applied in a wider range of predictive tasks.

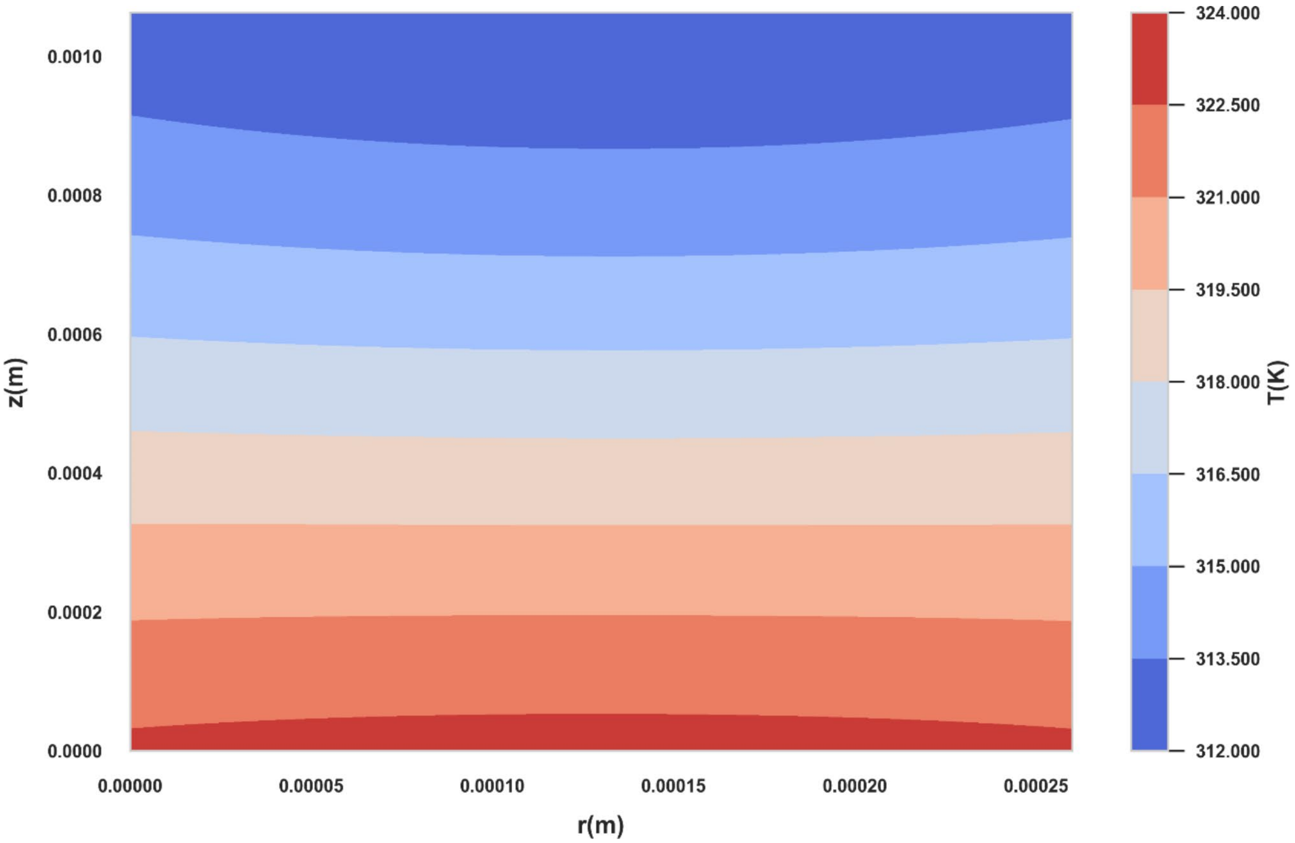

Fig. 7. Two-dimensional distribution of T (SVM model).

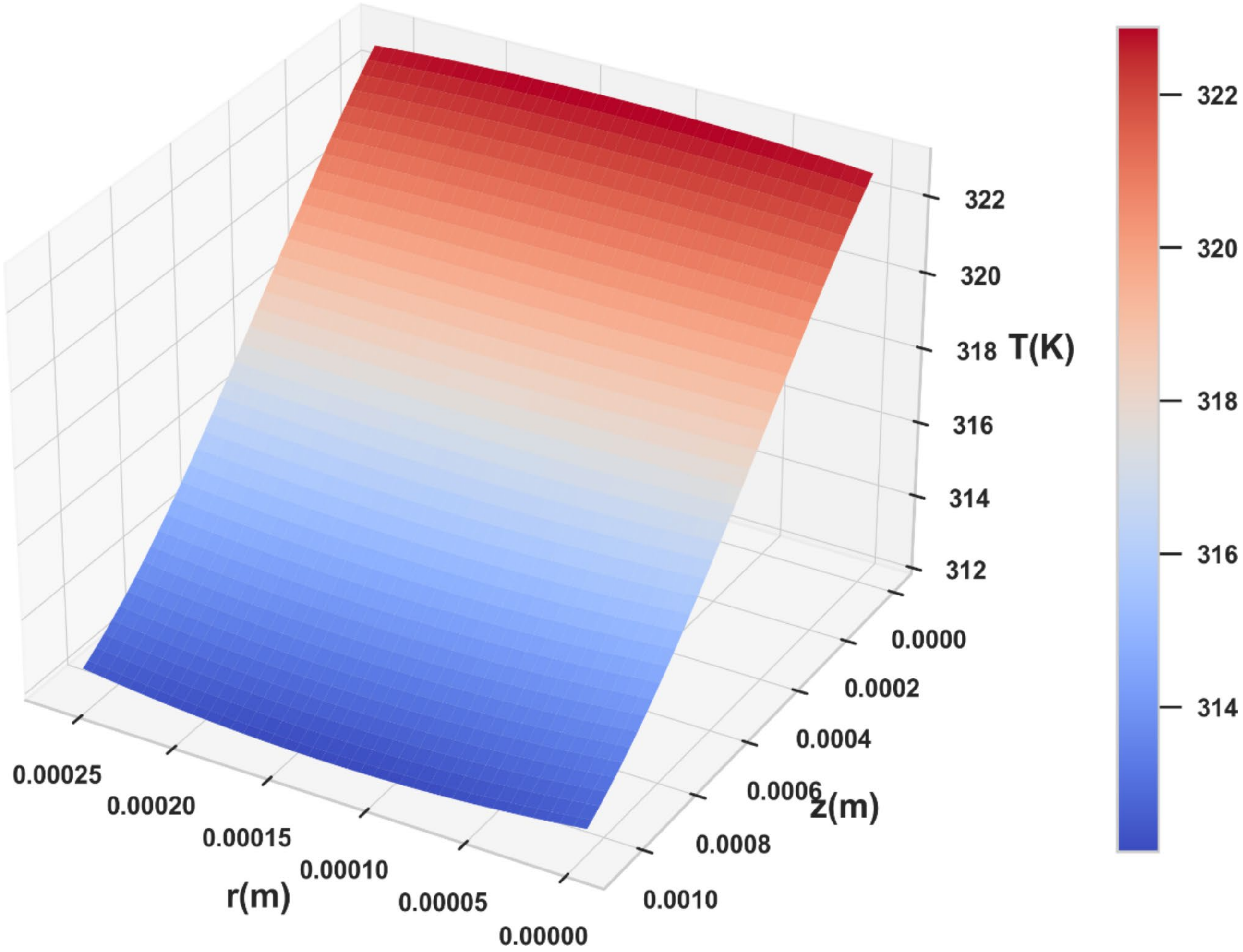

Fig. 8. 3D visualization of temperature as a function of coordinates.

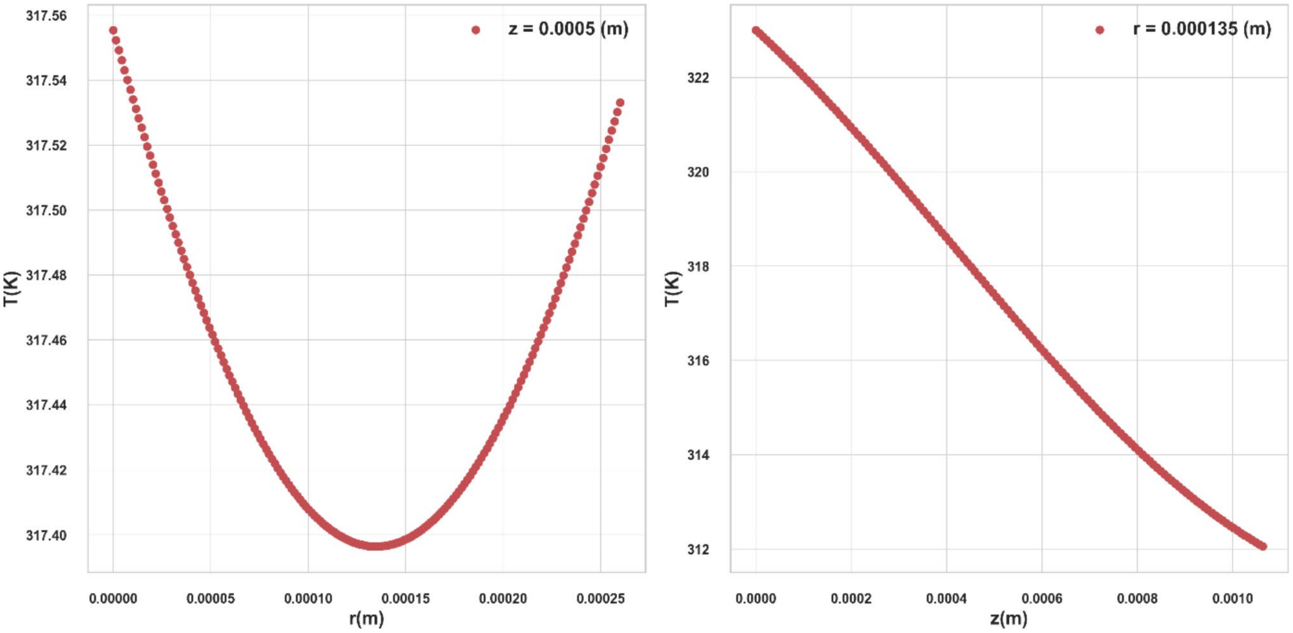

Fig. 9. T profiles in radial and axial directions.

Data availability

The data supporting this study are available when reasonably requested from the corresponding author.

Received: 4 July 2024; Accepted: 1 October 2024
Published online: 15 October 2024

References

1. Lv, Y. T. et al. In situ membrane separation drives nitrous oxide enrichment from nitrate denitrification for energy recovery. *J. Water Process. Eng.* **59**, 105064 (2024).
2. Ren, K. et al. Fabrication of hollow fiber composite membranes via opposite transmission reaction method for dye/salt separation. *J. Hazard. Mater.* **475**, 134856 (2024).
3. Xu, L. et al. Heterogeneous wettability membrane for efficient demulsification and separation of oil-in-water emulsions. *Chem. Eng. J.* **489**, 151466 (2024).
4. Richard, S. et al. Power-to-ammonia synthesis process with membrane reactors: Techno- economic study. *Int. J. Hydrog. Energy* **73**, 462–474 (2024).
5. Shigwan, P. & Padhiyar, N. A comparative study of membrane reactor and cylindrical reactor using multi-objective optimization for methane reforming process. *Int. J. Hydrog. Energy* **77**, 639–651 (2024).
6. Guo, Q. et al. Enhancement and optimization of membrane distillation processes: A systematic review of influential mechanisms, optimization and applications. *Desalination* **586**, 117862 (2024).
7. Zuo, L. et al. Characterization and prediction modeling of membrane distillation enhanced disc solar still. *J. Clean. Prod.* **449**, 141742 (2024).
8. Ismael, B. H. et al. Permeation flux prediction of vacuum membrane distillation using hybrid machine learning techniques. *Membranes* **13**(12), 900 (2023).
9. Aytaç, E. & Khayet, M. A deep dive into membrane distillation literature with data analysis, bibliometric methods, and machine learning. *Desalination* **553**, 116482 (2023).
10. Subrahmanya, T. M. et al. Self-surface heating membrane distillation for sustainable production of freshwater: A state of the art overview. *Prog. Mater. Sci.* **145**, 101309 (2024).
11. Zhu, Y. et al. Review of ammonia recovery and removal from wastewater using hydrophobic membrane distillation and membrane contactor. *Sep. Purif. Technol.* **328**, 125094 (2024).
12. Baghel, R. et al. CFD modeling of vacuum membrane distillation for removal of Naphthol blue black dye from aqueous solution using COMSOL multiphysics. *Chem. Eng. Res. Des.* **158**, 77–88 (2020).
13. FarisAbadi, A., Kazemeini, M. & Ekramipooya, A. Investigating a HEX membrane reactor for CO_2methanation using a Ni/ Al_2O_3catalyst: A CFD study. *Int. J. Hydrog. Energy* **48**(64), 25075–25091 (2023).
14. Ye, B. & Zhou, W. Efficiency increment of CFD modeling by using ANFIS artificial intelligence for thermal-based separation modeling. *Case Stud. Therm. Eng.* **60**, 104820 (2024).
15. Obaidullah, A. J. & Almehizia, A. A. Modeling and validation of purification of pharmaceutical compounds via hybrid processing of vacuum membrane distillation. *Sci. Rep.* **14**(1), 20734 (2024).
16. Wu, B. et al. Removal of 1,1,1-trichloroethane from water using a polyvinylidene fluoride hollow fiber membrane module: Vacuum membrane distillation operation. *Sep. Purif. Technol.* **52**(2), 301–309 (2006).
17. Tahvildari, K. et al. Numerical simulation studies on heat and mass transfer using vacuum membrane distillation. *Polym. Eng. Sci.* **54**(11), 2553–2559 (2014).
18. Zhang, F. *Cross-validation and Regression Analysis in High-Dimensional Sparse Linear Models* (Stanford University, 2011).
19. Usman, N., Utami, E. & Hartanto, A. D. Comparative Analysis of elliptic envelope, isolation forest, one-class SVM, and local outlier factor in detecting earthquakes with status anomaly using outlier. In *International Conference on Computer Science, Information Technology and Engineering (ICCoSITE)* (IEEE, 2023).
20. Storn, R. & Price, K. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *J. Global Optim.* **11**, 341–359 (1997).
21. Kumar, B. V., Oliva, D. & Suganthan, P. N. *Differential Evolution: From Theory to Practice* Vol. 1009 (Springer, 2022).
22. García–Nieto, P. J. et al. Hybrid DE optimised kernel SVR–relied techniques to forecast the outlet turbidity and outlet dissolved oxygen in distinct filtration media and micro-irrigation filters. *Biosyst. Eng.* **243**, 42–56 (2024).
23. Chakraborty, U. K. *Advances in Differential Evolution* Vol. 143 (Springer Science & Business Media, 2008).
24. Uludağ, G. & Uyar, A. Ş. Fitness landscape analysis of differential evolution algorithms. In *Fifth International Conference on Soft Computing, Computing with Words and Perceptions in System Analysis, Decision and Control* (IEEE, 2009).
25. Bishop, C. M. & Nasrabadi, N. M. *Pattern Recognition and Machine Learning* Vol. 4 (Springer, 2006).
26. Williams, P. M. Bayesian regularization and pruning using a Laplace prior. *Neural Comput.* **7**(1), 117–143 (1995).
27. Kruschke, J. K. Bayesian data analysis. *Wiley Interdiscip. Rev. Cogn. Sci.* **1**(5), 658–676 (2010).
28. Geurts, P., Ernst, D. & Wehenkel, L. Extremely randomized trees. *Mach. Learn.* **63**(1), 3–42 (2006).
29. Kocev, D., Ceci, M. & Stepišnik, T. Ensembles of extremely randomized predictive clustering trees for predicting structured outputs. *Mach. Learn.* **109**, 2213–2241 (2020).
30. Tay, J. K., Narasimhan, B. & Hastie, T. Elastic net regularization paths for all generalized linear models. *J. Stat. Softw.* 106 (2023).
31. Sanejouand, Y. H. Elastic network models: Theoretical and empirical foundations. *Biomol. Simul.* 601–616. (2013).
32. Zou, H. & Hastie, T. Regularization and variable selection via the elastic net. *J. Royal Stat. Soc. Ser. B Stat. Methodol.* **67**(2), 301–320 (2005).
33. Meyer, D., Leisch, F. & Hornik, K. The support vector machine under test. *Neurocomputing* **55**(1–2), 169–186 (2003).
34. Vapnik, V., Golowich, S. & Smola, A. Support vector method for function approximation, regression estimation and signal processing. *Adv. Neural Inf. Process. Syst.* **9** (1996).
35. Begum, M. Y. Advanced modeling based on machine learning for evaluation of drug nanoparticle preparation via green technology: Theoretical assessment of solubility variations. *Case Stud. Therm. Eng.* **45**, 103029 (2023).
36. Karatzoglou, A., Meyer, D. & Hornik, K. Support vector machines in R. *J. Stat. Softw.* **15**, 1–28 (2006).
37. Schölkopf, B., Smola, A. J. & Bach, F. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond* (MIT Press, 2002).

Author contributions

Y.W.: Conceptualization, Formal analysis, Investigation, Writing - Original Draft, Visualization.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to Y.W.

Reprints and permissions information is available at www.nature.com/reprints.

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