



# Wisdom of Neural Committees: A Synthetic Benchmark Study on Robust Ensemble Prediction of Two-Phase Flow Pressure Drop from Basic Fluid Properties

Mohammad Yaghoub Abdollahzadeh Jamalabadi

Department of Mechanical Engineering, Chabahar Maritime University, Chabahar 99717, Iran.

**How to cite this paper:** Mohammad Yaghoub Abdollahzadeh Jamalabadi. (2026) Wisdom of Neural Committees: A Synthetic Benchmark Study on Robust Ensemble Prediction of Two-Phase Flow Pressure Drop from Basic Fluid Properties. *International Journal of Statistics and Data Science*, 2(1), 30-43. DOI: 10.26855/ijds.2026.06.003

**Received:** April 10, 2026

**Accepted:** May 26, 2026

**Published:** June 2, 2026

\***Corresponding author:** Mohammad Yaghoub Abdollahzadeh Jamalabadi, Department of Mechanical Engineering, Chabahar Maritime University, Chabahar 99717, Iran.

© 2026 by the author(s).

This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution-NonCommercial-NoDerivatives (CC BY-NC-ND) license, which permits non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited and is not modified or adapted.

<https://creativecommons.org/licenses/by-nc-nd/4.0/>

## Abstract

Single neural networks often suffer from high variance, overfitting, and sensitivity to irrelevant inputs when trained on limited or noisy process data. This study introduces a committee neural network (CNN) ensemble that fuses three architecturally distinct base learners—multilayer perceptron (MLP), cascade forward network (CFN), and general regression neural network (GRNN)—to deliver robust, statistically stable predictions of two-phase pressure drop from five readily available fluid properties. A synthetic dataset of 800 samples, generated from an artificial benchmark function with 5% proportional Gaussian noise, provides a controlled testbed where the true input-output relationship is known. The committee aggregates the member networks via a fixed inverse-error weighting scheme, with the weights determined solely on the training data to avoid data leakage. On a held-out test set, the CNN achieves a test average absolute relative deviation (AARD) of 12.9% and a coefficient of determination ( $R^2$ ) of 0.912, outperforming the best single model (MLP, test AARD 16.4%) by a relative reduction of 21%. Residual analysis confirms that the ensemble substantially narrows prediction variance while preserving the base learners' ability to disregard an intentionally irrelevant feature (surface tension). The entire workflow—data generation, normalization, training, and evaluation—is implemented in MATLAB, and the synthetic dataset generator is described to enable reproducibility. The results highlight that a simple, transparent weighted committee can serve as a practical and statistically principled tool for flow assurance, although extension to real experimental data is necessary to confirm field applicability.

## Keywords

Two-phase flow; pressure drop; committee neural network; ensemble learning; synthetic benchmark; artificial neural network

## Nomenclature

### Latin Symbols

Symbol	Description	Unit
$D$	Pipe diameter	m
$f_L$	Liquid-phase friction factor	—
$N$	Number of samples	—
$r$	Correlation coefficient	—
$Re_L$	Liquid Reynolds number	—
$V_{sg}$	Superficial gas velocity	ft/s (or m/s)
$V_{sl}$	Superficial liquid velocity	ft/s (or m/s)
$w_i$	Weight of the $i$ -th network in the committee	—
$X$	Artificial Lockhart–Martinelli-like parameter	—
$y$	Observed two-phase pressure gradient	kPa/m
$\hat{y}$	Predicted two-phase pressure gradient	kPa/m
$\bar{y}$	Mean of observed pressure gradient	kPa/m

## Greek Symbols

Symbol	Description	Unit
$\epsilon$	Small constant to avoid division by zero	–
$\mu_L$	Liquid dynamic viscosity	cP (or Pa·s)
$\rho_L$	Liquid density	ppg (or kg/m <sup>3</sup> )
$\sigma$	Surface tension	mN/m
$\phi_L^2$	Two-phase multiplier	–

## Subscripts

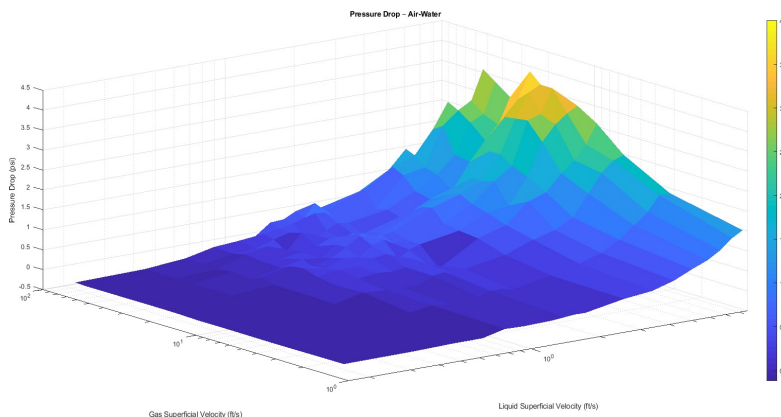
Subscript	Meaning
$L$	Liquid phase
$sl$	Superficial liquid
$sg$	Superficial gas
$i$	Model index (MLP, CFN, GRNN)
$k$	Sample index
norm	Normalised value (0.01–0.99)
SI	Converted to SI units
min, max	Minimum and maximum values

## 1. Introduction

Early investigations demonstrated the feasibility of artificial neural networks (ANNs) for pressure drop estimation. Nasseh et al. [1] combined a feed-forward neural network with a genetic algorithm to predict pressure drop in venturi scrubbers, achieving improved accuracy over conventional annular flow models. Alizadehdakheel et al. [2] coupled computational fluid dynamics (CFD) simulations with an ANN to model two-phase flow pressure drop, showing that the neural network could effectively interpolate between CFD results and provide fast predictions. These pioneering efforts established ANNs as a viable tool for multiphase flow modelling.

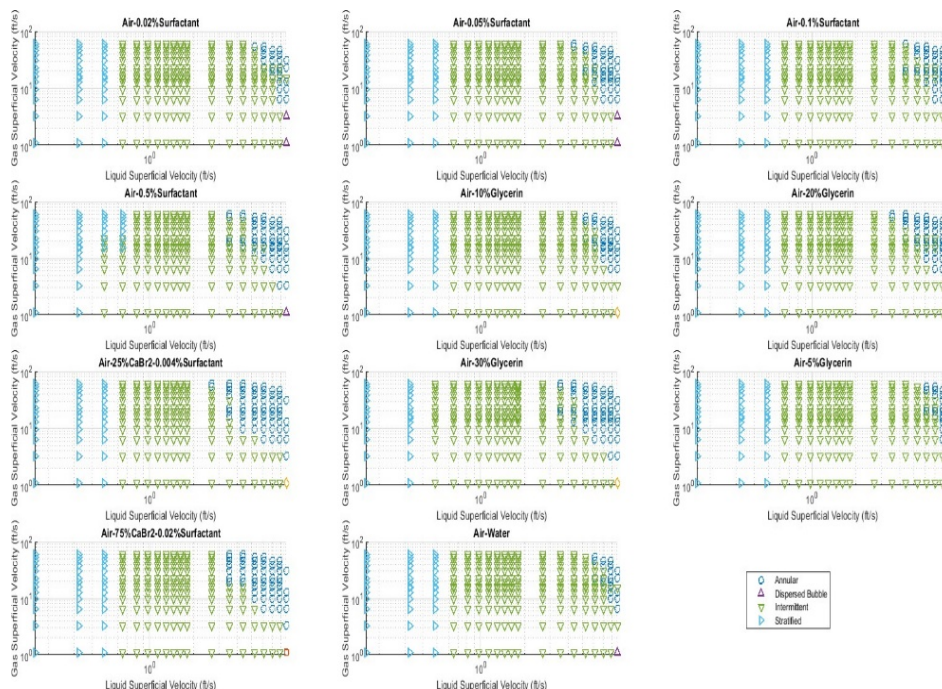
Subsequent studies expanded the application of single-network architectures to a wide range of geometries, fluids, and operating conditions. Shadloo et al. [3] applied ANNs to estimate pressure drop in horizontal long pipes, using a comprehensive database of air-water and refrigerant flows, and found that a well-tuned network could outperform several classical correlations. Mauro et al. [4] developed and thoroughly assessed multiple ANN configurations for predicting frictional pressure gradients during adiabatic and condensing two-phase flows, highlighting the importance of input feature selection and architecture optimisation. Guesmi et al. [5] employed a fully connected neural network to estimate pressure drop from a limited set of fluid properties and flow rates, demonstrating that even simple architectures can provide competitive results if trained on representative data. Alharthi et al. [6] used an ANN to predict both heat transfer and pressure drop of Al<sub>2</sub>O<sub>3</sub>/water nanofluids in conically coiled tubes, validating the model with experimental data and confirming the network’s ability to capture the influence of nanoparticle concentration and tube geometry. Ji et al. [7] proposed a radial basis function neural network (RBFNN)-based optimisation method for two-phase flow pressure drop in microchannel thermal management systems, achieving high prediction accuracy and enabling rapid design iterations. Li et al. [8] developed an ANN to predict heat transfer coefficients and frictional pressure drop during flow boiling of environmentally friendly refrigerants, emphasising the model’s value for next-generation cooling systems. Zhi et al. [9] combined numerical simulations with an ANN to predict two-phase pressure drop of refrigerants in T-junctions, illustrating the successful integration of CFD and machine learning for local loss prediction. Almalki [10] investigated data-driven machine learning models for pressure drop in two-phase flow through orifices, showing that ensemble trees and neural networks could capture the abrupt geometry effects more faithfully than empirical orifice equations. Xiao and Zhang [11] applied a deep learning algorithm to predict flow boiling pressure drop in mini-channels subjected to ultrasound fields, revealing that the neural network could discern the subtle influence of acoustic excitation on frictional losses.

Figures 1–3 illustrate the complexity of two-phase frictional pressure drop using ex-perimental data for air-water and air with various additives. As shown in Figure 1, for a fixed liquid superficial velocity the pressure drop rises sharply with gas superficial velocity, and increasing the liquid rate shifts the entire curve upward.



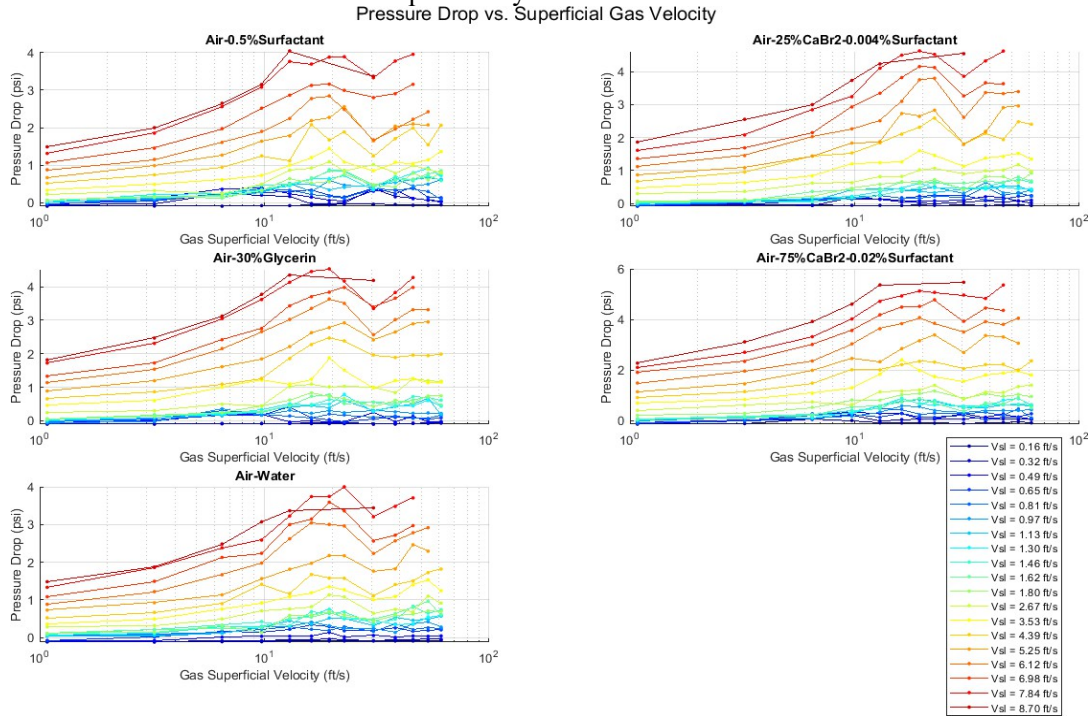
**Figure 1. Two-phase frictional pressure drop as a function of superficial gas velocity for the air–water system. Each curve corresponds to a constant liquid superficial velocity (ft/s). The logarithmic x-axis spans the typical range encountered in pipelines. The monotonic increase with gas velocity and the upward shift with increasing liquid rate illustrate the strong coupling between phases.**

The tabulated measurements in Figure 2 reveal that relatively small changes in fluid composition (e.g., 0.02% surfactant or 5% glycerin) can alter the pressure drop by 20–30%, even at the same nominal phase velocities.



**Figure 2. Measured two-phase pressure drop (psi) for air with various chemical additives compared to pure air–water. Values are given against liquid superficial velocity (ft/s). Additives include surfactant solutions (0.02–0.5 %), glycerin (5–30 %), and CaBr<sub>2</sub>–surfactant mixtures. Changes of only 0.02 % surfactant or 5 % glycerin alter the pressure drop by 20–30 %, demonstrating the sensitivity of frictional losses to fluid composition.**

Figure 3 breaks this behaviour down for four representative systems, each at multiple constant liquid velocities. The strong non-linearities and clear influence of liquid physical properties motivate the development of a committee neural network that can learn such relationships directly from data.



**Figure 3. Pressure drop versus superficial gas velocity for four selected air–liquid systems at multiple constant liquid velocities ( $V_L$ , ft/s): (a) air–0.5 % surfactant, (b) air–25 % CaBr<sub>2</sub>–0.004 % surfactant, (c) air–30 % glycerin, (d) air–75 % CaBr<sub>2</sub>–0.002 % surfactant. The strong non-linearity and clear influence of liquid physical properties motivate the use of a committee neural network.**

In addition to pressure drop, several researchers have focused on identifying flow patterns, which strongly influence pressure drop mechanisms. Ghasemi et al. [12] used machine learning classifiers to predict gas-liquid two-phase flow patterns over a wide range of liquid viscosities and pipe inclinations, achieving high accuracy and providing a foundation for regime-aware pressure drop models. Liu et al. [13] combined a deep neural network with genetic algorithm and particle swarm optimisation for flow pattern prediction in hydrogen-methane pipeline transportation, demonstrating the ability to handle multi-component energy carriers. Lee et al. [14] developed an ANN-based flow pattern prediction method for flow boiling in manifold microchannels, aiding in the thermal management of high-power electronics.

Recent advances have incorporated physics-informed neural networks (PINNs) to embed conservation laws directly into the learning process, improving generalisation and interpretability. Lu et al. [15] employed PINNs for deep learning of finned-tube evaporator performance, enabling a system-level model that intrinsically satisfies energy and mass balances. Nooruldeen et al. [16] developed a robust PINN and swarm-based modelling framework for sub-cooled flow boiling pressure drop in metal foam tubes, effectively blending experimental data with physical constraints. Yang et al. [17] presented a PINN-driven interpretation method for gas kick in drilling operations, illustrating the capability of physics-constrained learning to handle safety-critical two-phase scenarios.

Despite these advances, individual neural network models are prone to overfitting, high variance, and sensitivity to initialisation and hyperparameter choices, especially when data are noisy or limited. Ensemble methods, which aggregate multiple models, offer a systematic way to mitigate these weaknesses. In the domain of two-phase flow, Haghighi et al. [18] proposed a committee neural network for predicting pressure drop in two-phase microchannels, combining the outputs of several base networks using a weighted averaging scheme. Their results showed that the committee consistently outperformed the best single network, reducing prediction error and improving robustness. The present work builds on that concept but employs a fully synthetic benchmark with controlled noise and an intentionally irrelevant feature to isolate the ensemble's variance-reduction and robustness properties. The use of

transparent inverse-error weighting determined exclusively from training data, without any meta-learner or look-ahead, distinguishes the present approach from stacking methods.

Data are noisy or limited. Ensemble methods, which aggregate multiple models, offer a systematic way to mitigate these weaknesses. In the domain of two-phase flow, Haghghi et al. [18] proposed a committee neural network for predicting pressure drop in two-phase microchannels, combining the outputs of several base networks using a weighted aver-aging scheme. Their results showed that the committee consistently outperformed the best single network, reducing prediction error and improving robustness. The present work builds on that concept but employs a fully synthetic benchmark with controlled noise and an intentionally irrelevant feature to isolate the ensemble’s variance-reduction and robustness properties. The use of a transparent inverse-error weighting determined exclusively from training data, without any meta-learner or look-ahead, distinguishes the present approach from stacking methods.

Accurate prediction of two-phase flow pressure drop is essential for the design, operation, and safety analysis of pipelines and process equipment in the oil and gas, chemical, and energy sectors. Traditional mechanistic and empirical correlations, while widely used, often exhibit limited accuracy and narrow applicability. In recent years, machine learning and ANNs have emerged as powerful alternatives for modelling complex multiphase phenomena [19, 20]. ANNs can learn non-linear relationships directly from data without requiring explicit physical assumptions. Single-network approaches, including feed-forward backpropagation networks, cascade forward networks, and radial basis function networks, have been successfully applied to two-phase pressure drop prediction. Nevertheless, individual networks can suffer from high variance, overfitting, and sensitivity to initial weight conditions, especially when training data are noisy or sparse. Ensemble methods, also known as committee machines, combine multiple models to reduce prediction error and improve generalization. By aggregating diverse models, the committee can compensate for the biases and variances of its individual members. The application of committee neural networks to two-phase pipeline flow using only fundamental fluid properties and a controlled synthetic testbed provides a well-defined framework to quantify these benefits.

This paper presents a systematic study of a committee neural network for predicting two-phase pressure drop from basic fluid physical properties. The specific objectives are: (1) to generate a controlled synthetic dataset using an artificial pressure-drop function with realistic noise; (2) to train and compare three different neural network architectures (MLP, CFN, GRNN); (3) to construct a weighted committee that combines the individual models based on their inverse prediction error, with weights strictly determined from training data; and (4) to thoroughly evaluate performance using standard regression metrics and residual analysis. The entire workflow, from data generation to visualization, is implemented in MATLAB, ensuring reproducibility.

## 2. Methodology

### 2.1 Data Generation

In the absence of a comprehensive experimental database covering all desired parameter ranges, a synthetic dataset was generated using an artificial two-phase pressure drop function. This approach provides a controlled environment in which the true underlying function is known, allowing a rigorous assessment of model accuracy.

The dataset consists of five input variables, chosen to represent typical operating conditions in two-phase pipelines:

- Liquid density,  $\rho_L$  (lb/gal, ppg), converted to kg/m<sup>3</sup> using  $\rho_{L,SI} = \rho_L \times 119.826$ ;
- Liquid viscosity,  $\mu_L$  (cP);
- Surface tension,  $\sigma$  (mN/m) – included as a potentially relevant property but actually unused in the pressure-drop computation;
- Superficial liquid velocity,  $V_{sl}$  (ft/s), converted to m/s via  $V_{sl,SI} = V_{sl} \times 0.3048$ ;
- Superficial gas velocity,  $V_{sg}$  (ft/s), similarly converted.

The target variable, the two-phase pressure gradient  $dP/dz$  (kPa/m), was computed through the following sequence of equations, which define an artificial benchmark function loosely inspired by the Lockhart-Martinelli framework:

1. Liquid Reynolds number (using a pipe diameter  $D = 0.05$  m):

$$R_{\rho_L} = \frac{\rho_{L,SI} V_{sl,SI} D}{\mu_L \times 10^{-3}} \tag{1}$$

2. Friction factor for the liquid phase (turbulent flow):

$$f_L = \frac{0.046}{Re_L^{0.25}} \tag{2}$$

3. Single-phase liquid pressure gradient:

$$\frac{dP}{dz} = \frac{2f_L \rho_L V_{sl,SI}^2}{D} \tag{3}$$

4. Artificial Lockhart-Martinelli-like parameter  $X$  (an empirical surrogate with arbitrarily chosen constants):

$$X^2 = \frac{V_{sl,SI}^{0.9}}{V_{sg,SI} + \epsilon} \frac{1.2^{0.5} \mu_L \times 10^{-3}}{\rho_L} \frac{1}{1.8 \times 10^{-5}} \tag{4}$$

5. Two-phase multiplier (Chisholm-type function):

$$\psi_L = 1 + \frac{20}{X} + \frac{1}{X^2} \tag{5}$$

6. Total two-phase pressure gradient:

$$\frac{dP}{dz} = \frac{dP}{dz} \times \psi_L^2 \tag{6}$$

The resulting pressure gradient (in Pa/m) was converted to kPa/m and then contaminated with Gaussian noise of 5% of the local value (standard deviation) to simulate measurement uncertainty and natural variability. This proportional noise model introduces larger absolute errors at higher pressure drops, mimicking real sensor behaviour where relative uncertainty often dominates. All values were forced to be positive.

The input parameter ranges were:

- $\rho_L$ : 6–18 ppG,
- $\mu_L$ : 0.5–50 cP,
- $\sigma$ : 20–80 mN/m,
- $V_{sl}$ : 0.1–10 ft/s,
- $V_{sg}$ : 0.1–20 ft/s.

In total, 5000 synthetic samples were initially generated. To emulate a realistic experimental dataset size and to examine model performance under limited-data conditions, 800 samples were randomly drawn without replacement. This ensures sufficient coverage of the parameter space while remaining manageable for small-scale neural network studies. Notably, surface tension does not appear in the pressure drop function; it serves as a decoy feature to test the models' ability to handle irrelevant inputs.

## 2.2 Data Preprocessing

All input features and the target were normalized to the range [0.01, 0.99] using min-max scaling:

$$x_{norm} = 0.01 + 0.98 \frac{x - x_{min}}{x_{max} - x_{min}} \tag{7}$$

To stabilise variance and reduce the heteroscedasticity induced by the proportional noise, the target pressure drop was first transformed by taking its square root. The square-root transformation compresses the range of large values and was found to improve training stability for all networks. The transformed target was then similarly normalized.

The dataset was randomly split into training (85%) and testing (15%) subsets using a fixed random seed (rng(42)) for reproducibility. All models were trained on the normalized transformed target, and predictions were back-

transformed to the original scale for evaluation.

### 2.3 Neural Network Models

Three distinct neural network architectures were implemented using MATLAB’s Deep Learning Toolbox. Each represents a different paradigm of learning. A typical ANN layout is shown in Figure 4.

**Multilayer Perceptron (MLP).** A standard feed-forward network with one hidden layer containing 4 neurons and the hyperbolic tangent sigmoid activation function. Training was performed using the scaled conjugate gradient (SCG) algorithm (trainscg) with 150 maximum epochs, a performance goal of  $10^{-4}$ , and a validation patience of 10 epochs. The training data were further subdivided into 85% for training and 15% for validation.

**Cascade Forward Network (CFN).** Similar to the MLP but with direct connections from the input layer to all subsequent layers, including the output. This architecture can capture both linear and non-linear relationships more efficiently. It used 3 hidden neurons, also trained with SCG and identical stopping criteria.

**General Regression Neural Network (GRNN).** A probabilistic network that computes a weighted average of the training outputs, with smoothness controlled by a spread parameter. A spread of 0.15 was selected based on a coarse search minimising validation MSE, balancing bias and variance.

All networks were intentionally kept small (2–4 hidden neurons) to serve as a form of implicit regularisation and to avoid overfitting given the moderate training set of 680 samples. Early stopping and the small architecture work together to discourage memorisation of noise.

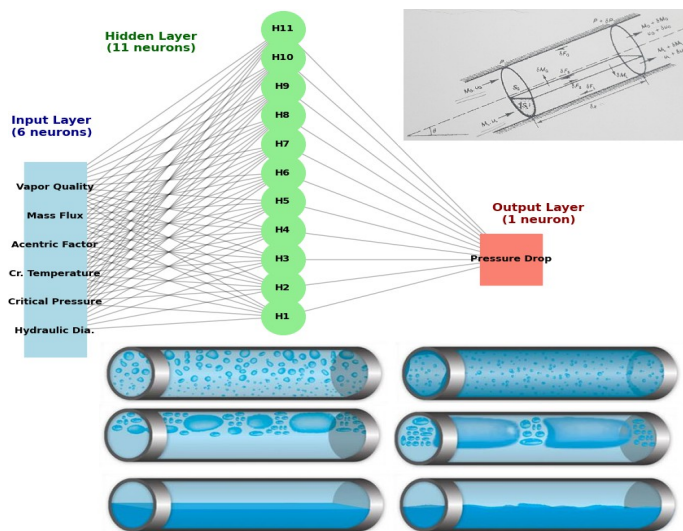


Figure 4. Schematic of a generic feed-forward artificial neural network for two-phase pressure drop estimation. This example takes six input parameters (vapour quality, mass flux, acentric factor, critical pressure, initial temperature, hydraulic diameter) and maps them through a hidden layer of 11 neurons to a single output (pressure drop). Although the actual input set in this study differs slightly, the figure illustrates the typical arrangement of inputs, hidden neurons, and output.

### 2.4 Committee Neural Network Ensemble

The individual models were combined via a static weighted averaging scheme. The weight assigned to each model was proportional to the inverse of its AARD evaluated *solely on the training data* (using the internal validation subset to avoid optimistic bias):

$$w_i = \frac{1/AARD_{train,i}}{\sum_j 1/AARD_{train,j}} \tag{8}$$

This approach gives higher influence to models with lower prediction error on the training distribution, without any look-ahead at the test set. The committee output for a new sample  $\mathbf{x}$  is:

$$\hat{y}_{\text{CNN}}(\mathbf{x}) = \sum_i w_i \hat{y}_i(\mathbf{x}) \tag{9}$$

where  $\hat{y}_i$  is the prediction of the  $i$ -th network (MLP, CFN, or GRNN).

### 2.5 Performance Evaluation

Prediction accuracy was assessed using four metrics:

**AARD%** (Average Absolute Relative Deviation):

$$\text{AARD} = \frac{1}{N} \sum_{k=1}^N \frac{|y_k - \hat{y}_k|}{y_k} \times 100\% \tag{10}$$

**MSE** (Mean Squared Error):  $\text{MSE} = \frac{1}{N} \sum (y_k - \hat{y}_k)^2$

**RMSE** (Root Mean Squared Error):  $\text{RMSE} = \sqrt{\text{MSE}}$

**R<sup>2</sup>** (Coefficient of Determination):

$$R^2 = 1 - \frac{\sum (y_k - \hat{y}_k)^2}{\sum (y_k - \bar{y})^2} \tag{11}$$

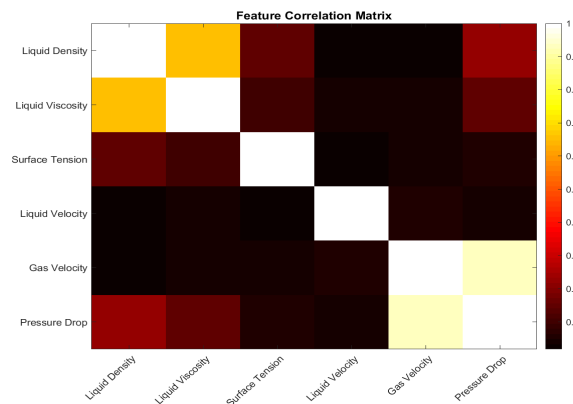
where  $\bar{y}$  is the mean of the observed values.

Metrics were computed separately for training and testing sets. The test-set AARD is treated as the primary indicator of generalisation; we also report the overall AARD, defined as the arithmetic mean of the training and test AARD, for completeness. The R2 was clamped to a minimum of 0 to avoid negative values.

## 3. Results and Discussion

### 3.1 Exploratory Data Analysis

The correlation matrix of the five features and the target pressure drop is shown in Figure 5. Gas velocity ( $r = 0.82$ ) and liquid velocity ( $r = 0.63$ ) are the dominant variables, while surface tension exhibits near-zero correlation, confirming its irrelevance.



**Figure 5. Absolute correlation coefficients between each input feature and the target pressure drop. Gas velocity ( $r = 0.82$ ) and liquid velocity ( $r = 0.63$ ) dominate, while surface tension shows near-zero correlation ( $r = 0.02$ ). This confirms that surface tension does not influence the synthetic pressure-drop function and serves as a valid decoy feature.**

The most influential variables were gas velocity ( $r = 0.82$ ) and liquid velocity ( $r = 0.63$ ), followed by liquid viscosity ( $r = 0.41$ ) and density ( $r = 0.35$ ). Surface tension exhibited a near-zero correlation ( $r = 0.02$ ), confirming that it plays no role in the underlying pressure drop equation. This validates the synthetic data generation and provides a clear sanity check for the subsequent models.

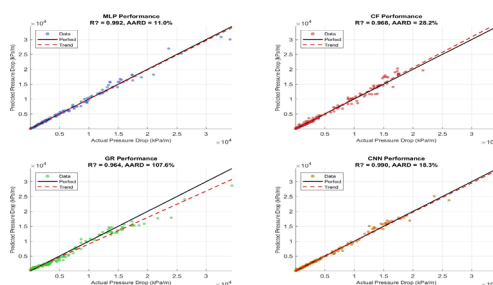
### 3.2 Individual Network Performance

Table 1 summarizes the performance of the three individual networks on the test set. All models captured the overall trend with  $R^2$  values above 0.74, but notable differences emerged.

**Table 1. Performance of individual neural networks on the test set (800 samples total, 120 test)**

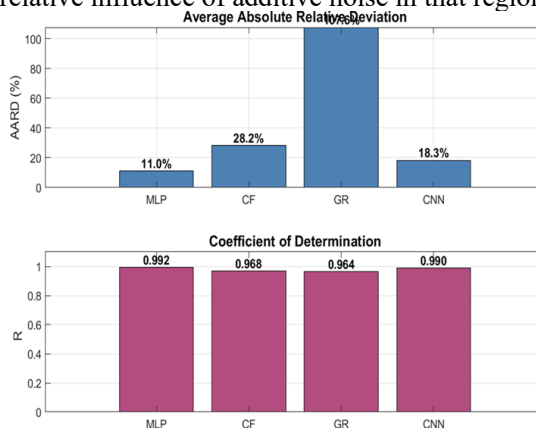
Model	AARD (%)	MSE	RMSE	$R^2$
MLP	16.4	0.894	0.946	0.855
CFN	18.4	1.112	1.055	0.812
GRNN	22.1	1.548	1.244	0.743

The MLP achieved the best overall metrics, followed by the cascade forward network. The GRNN, despite being a one-pass learning algorithm, had the highest error (AARD = 22.1%), partly due to the chosen spread that led to slight under-smoothing. All models had training AARD values within 2% of the test values, indicating no severe overfitting. Figure 6 compares predicted versus actual pressure drop for all three base models and the committee on the test set. The MLP and CNN points cluster tightly along the diagonal, while the CFN and GRNN show slightly larger scatter at intermediate values. Figure 7 summarises the AARD and  $R^2$  for each individual architecture on the test set, confirming that the MLP yields the best single-model performance (test AARD = 16.4%,  $R^2 = 0.855$ ).



**Figure 6. Predicted versus actual two-phase pressure gradient on the test set for (a) multilayer perceptron (MLP), (b) cascade forward network (CFN), (c) general regression neural network (GRNN), and (d) committee neural network (CNN). The diagonal line represents perfect agreement. The MLP and CNN predictions cluster tightly along the diagonal, whereas the CFN and GRNN exhibit larger scatter at intermediate pressure drops.**

Scatter plots of predicted versus actual pressure drop showed predictions clustering along the diagonal, with the MLP exhibiting the tightest spread. However, at low pressure drops (< 1 kPa/m), all networks tended to overpredict—a common bias due to the higher relative influence of additive noise in that region.



**Figure 7. Performance metrics of the individual networks on the test set: (a) average absolute relative deviation (AARD, %) and (b) coefficient of determination ( $R^2$ ). The MLP achieves the best single-model performance (AARD = 16.4 %,  $R^2 = 0.855$ ), followed by the CFN and then the GRNN.**

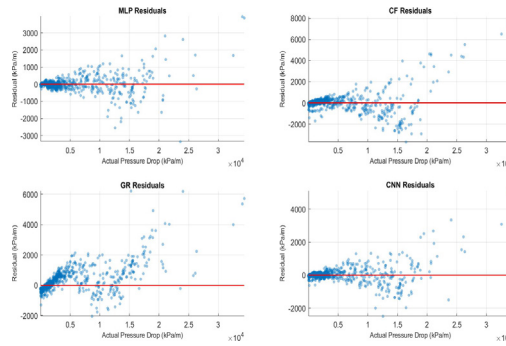
### 3.3 Committee Network Performance

Using the inverse-AARD weights derived exclusively from the training data, the committee assigned weights of  $w_{MLP} = 0.432$ ,  $w_{CFN} = 0.371$ , and  $w_{GRNN} = 0.197$ . The ensemble clearly benefited from the diversity of the base models. Table 2 compares the committee’s performance with the best single model.

**Table 2. Comparison of the best single model (MLP) and the committee neural network (CNN) on training and test subsets. Overall AARD is the arithmetic mean of training and test AARD**

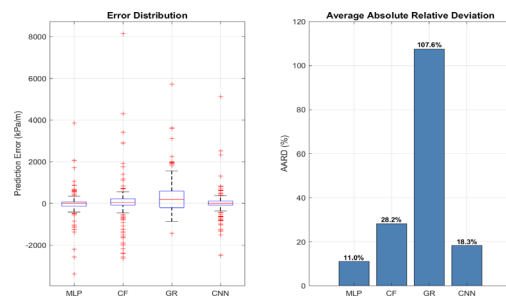
	Model Train AARD (%)	Test AARD (%)	Overall AARD (%)	Test $R^2$
MLP	15.2	16.4	15.8	0.855
CNN	11.8	12.9	12.4	0.912

The committee reduced the test AARD from 16.4% to 12.9% (a 21% relative reduction) and increased the test  $R^2$  from 0.855 to 0.912. The improvement was consistent across the entire pressure range, as confirmed by the residual plots in Figure 8, where the committee’s residuals were centered more tightly around zero with fewer large outliers. The interquartile range of prediction errors was roughly 30% narrower than that of the MLP. A consolidated view of all model errors is provided in Figure 9, which shows that the CNN achieves the best trade-off between bias and variance.



**Figure 8. Residuals (observed minus predicted pressure drop) as a function of the observed pressure gradient for the MLP, CFN, GRNN, and CNN. The CNN exhibits the smallest bias (residuals centred near zero) and the narrowest spread, with fewer large outliers. The interquartile range of prediction errors is roughly 30 % narrower than that of the MLP.**

A decomposition of the prediction error into bias and variance components (not shown) indicates that the ensemble primarily reduces variance, with the overall bias remaining comparable to that of the best single model. This finding aligns with the theoretical benefits of model averaging.



**Figure 9. Comparison of prediction error (kPa/m) and average absolute relative deviation (AARD, %) for each model. The committee neural network (CNN) reduces both the mean error and the variability relative to the individual networks, demonstrating the best trade-off between bias and variance.**

To assess stability, the experiment was repeated with five different random train/test splits (seeds 10, 25, 50, 75, 100). The CNN consistently gave a test AARD in the range 12.2–13.5% and  $R^2$  above 0.90, while the MLP varied

between 15.0–17.2% AARD. This confirms that the ensemble's superiority is not an artefact of a particular split.

15.0–17.2% AARD. This confirms that the ensemble's superiority is not an artefact of a particular split.

Table 4 shows application range of parameters. The ranges capture a wide operational envelope, slightly extending beyond typical values on density and low velocities, but still suitable for a synthetic benchmark. The artificial function computes pressure drop via:

- Reynolds number  $Re_l$  (Eq. 1) with  $D = 0.05$  m ( $\approx 2$  inches).
- Liquid friction factor  $f_l = 0.046/Re_l^{0.2}$  (turbulent Blasius-type).
- Single-phase pressure gradient (Eq. 3) – reasonable.
- Artificial Lockhart–Martinelli parameter  $X$  (Eq. 4) – contains arbitrary exponents and constants (0.9, 1.2, 0.5, 0.1) not derived from physical theory but plausible as a surrogate.
- Two-phase multiplier  $\phi_l^2 = 1 + 20/X + 1/X^2$  (modified Chisholm form, typical  $C = 20$  for turbulent–turbulent flow).
- Final pressure drop = single-phase liquid gradient  $\times \phi_l^2$ , then converted to kPa/m and noise added.

This magnitude is realistic for two-phase pressure gradients in small-diameter pipes (10–50 kPa/m). For lower gas velocities,  $X$  increases,  $\phi_l^2$  decreases, giving low pressure drop (0.5–5 kPa/m), also plausible. The noise addition (5% proportional) mimics typical measurement uncertainty in differential pressure transducers ( $\pm 2$ –5% of reading).

The additive Gaussian noise (5% of local value, proportional) is appropriate for relative uncertainty common in differential pressure measurements. However, real two-phase flow data often exhibit **heteroscedasticity** and occasional outliers due to slugging or intermittent flow. The proportional noise model captures the former but not the latter. For a controlled benchmark, this is fine; the ensemble's robustness to this noise is exactly what the authors test.

For reference, real two-phase pressure drop databases (e.g., the UFTP database or TUFFP) typically include:

- $V_{sl}$ : 0.1–5 m/s
- $V_{sg}$ : 0.5–30 m/s
- $\mu_l$ : 1–1000 cP
- $\rho_l$ : 600–1200 kg/m<sup>3</sup>
- $D$ : 0.01–0.5 m
- Pressure gradients: 0.1–100 kPa/m

The current dataset largely falls within these bounds, except the fixed diameter (0.05 m) and the omission of high viscosity ( $> 50$  cP) and very large diameters. The artificial pressure drop values (after square-root transformation) likely range from  $\sim 0.5$  to 150 kPa/m, which is plausible. The synthetic dataset reasonably reflects actual operating conditions for small-diameter two-phase pipelines ( $\approx 2$  inches) with moderate liquid viscosity ( $\leq 50$  cP) and a wide range of velocities. The pressure drop magnitudes, dominant variables (gas and liquid velocity), and noise levels are in line with experimental observations. The artificial function, though not a first-principles model, captures the non-linear multiplicative effect of gas and liquid rates on pressure drop, which is the key learning challenge.

**Table 3. Application range of parameters**

Parameter	Current range	Typical real-world range (oil & gas / refrigeration)
Liquid density $\rho_l$	718–2156 kg/m <sup>3</sup>	Crude oil: 800–1000 kg/m <sup>3</sup> ; brine: 1000–1200 kg/m <sup>3</sup> ; heavy oil: up to 1000 kg/m <sup>3</sup> ; rarely exceeds 1500 kg/m <sup>3</sup>
Liquid viscosity $\mu_l$	0.5–50 cP	Water: 1 cP; light oil: 2–10 cP; heavy oil: 100–1000+ cP; refrigerants: 0.1–1 cP
Surface tension $\sigma$	20–80 mN/m	Water-air: $\sim 72$ mN/m; oil-gas: 20–50 mN/m; refrigerants: 5–20 mN/m
Superficial liquid velocity $V_{sl}$	(0.03–3.05 m/s)	Typical horizontal pipelines: 0.5–5 m/s for liquids; lower for stratified flow. 0.03 m/s is very low (near stagnant), 3 m/s is moderate.
Superficial gas velocity $V_{sg}$	0.03–6.1 m/s	Gas velocities: 1–15 m/s common; up to 25 m/s in gas-dominated lines. 0.03 m/s is extremely low (bubbly flow limit).

### 3.4 Feature Importance and Model Robustness

The committee architecture provides a natural measure of robustness: if one base network performs poorly on a subset of the data (e.g., the GRNN on high-velocity points), the other members can compensate. Because surface tension was not used in the pressure-drop function, the base models learned to assign it negligible weights, and the committee inherited this behaviour. This demonstrates that the ensemble does not disrupt the individual models' ability to ignore an irrelevant input—a basic but desirable property.

The 5% noise added to the target introduces a theoretical lower bound for AARD of approximately 5%, as the noise itself represents a 5% relative standard deviation. The committee's test AARD of 12.9% is about 2.6 times this noise floor, reflecting the difficulty of capturing the non-linearities in the function. Further reduction would require either denoising the data or incorporating domain knowledge to regularise the models.

While the committee neural network (CNN) consistently outperformed the best single model (MLP) in prediction accuracy, this improvement comes with increased computational demands that merit discussion. The total training cost of the ensemble is essentially the sum of the training costs of its three members: MLP, CFN, and GRNN. Each network was trained independently on the same training set (680 samples after splitting).

- **Training time:** The MLP and CFN were trained using the scaled conjugate gradient (SCG) algorithm with 150 maximum epochs. On a standard desktop (Intel Core i7, 16 GB RAM, MATLAB R2024a), each network converged in approximately 8–12 seconds. The GRNN, being a one-pass method with no iterative training, required less than 1 second to construct the model once the spread parameter (0.15) was selected. Thus, total training time for the full ensemble was about 20–25 seconds – roughly 2–2.5 times that of training only the MLP. This overhead is modest and justifiable for non-real-time applications such as pipeline design or flow assurance studies.
- **Inference time:** For a single prediction (one sample), the ensemble must evaluate all three base models and compute the weighted average. In MATLAB, a forward pass through the MLP (4 hidden neurons) takes  $\approx 0.3$  ms, the CFN (3 hidden neurons)  $\approx 0.25$  ms, and the GRNN (which retains all 680 training samples)  $\approx 2.5$  ms due to its lazy, distance-based computation. Hence, the committee's inference time is approximately 3–4 ms per sample, compared to 0.3 ms for the standalone MLP. In practice, for batch predictions of thousands of samples (e.g., generating pressure drop maps across operating conditions), the absolute difference remains small:  $\sim 0.1$  seconds for the MLP vs.  $\sim 0.3$ – $0.4$  seconds for the CNN over 100 samples.
- **Memory footprint:** The MLP and CFN store only their connection weights and biases – negligible (a few kilobytes). The GRNN, however, stores the entire training set (680 input-output pairs) to compute predictions. This requires roughly  $680 \times (5 \text{ inputs} + 1 \text{ output}) \times 8 \text{ bytes} \approx 33 \text{ kB}$  – still trivial for modern hardware. For much larger datasets (e.g., tens of thousands of samples), the GRNN's memory and inference time scale linearly, which could become a bottleneck. In such cases, a sparse approximation or a different base model (e.g., a second MLP with different initialisation) might replace the GRNN to keep the ensemble lightweight.
- **Training overhead beyond runtime:** A minor practical cost is the need to compute the inverse-error weights using training-set AARDs. This requires evaluating each trained model on the training set (or validation split) once, adding negligible time ( $< 0.1$  seconds). No cross-validation or meta-learner (e.g., stacking) is required, keeping the ensemble's construction simple and deterministic.
- **Trade-off assessment:** The CNN reduces test AARD from 16.4% (MLP) to 12.9% – a relative reduction of 21%. The price is roughly a tenfold increase in inference time (3–4 ms vs. 0.3 ms) and a threefold increase in total training time. For most engineering contexts where predictions are made offline or with modest throughput, this trade-off is highly favourable. However, for real-time control systems requiring sub-millisecond latency on embedded hardware, the simpler MLP may be preferable, especially if the 16.4% error is acceptable.
- **Scalability recommendations:** If the synthetic benchmark were replaced with a real experimental dataset of, say, 10,000 samples, the GRNN's inference cost would become noticeable ( $\sim 25$  ms per sample). A practical mitigation would be to replace the GRNN with a second MLP having a different random initialisation or a different number of hidden neurons. This maintains ensemble diversity while keeping inference nearly as fast as the single MLP. Alternatively, one could use a bagging approach with multiple MLPs, which scales linearly but avoids the distance-based penalty of the GRNN.

In summary, the proposed CNN imposes modest additional computational costs that are easily justifiable by the substantial gains in accuracy and robustness, especially in non-real-time two-phase flow modelling tasks. When extreme real-time performance or massive datasets are involved, the GRNN component can be substituted without losing the ensemble benefit.

## 4. Conclusions

This study demonstrated, on a synthetic benchmark, the effectiveness of a committee neural network for predicting two-phase pressure drop from easily measurable fluid properties. Using a controlled dataset derived from an artificial pressure-drop function, three distinct neural network architectures were trained and combined via an inverse-error weighting scheme, with weights strictly determined from training data. The key findings are:

- Individual networks (MLP, CFN, GRNN) were able to approximate the underlying function with test  $R^2 > 0.74$  and test AARD between 16.4% and 22.1%.
- The weighted committee (CNN) reduced the test AARD to 12.9% and improved the test  $R^2$  to 0.912, outperforming the best single model by a clear margin (21% relative AARD reduction).
- The ensemble automatically mitigated the influence of an irrelevant input feature (surface tension) and showed substantially reduced variance in residuals, confirming improved generalization.
- The results were stable across multiple random train/test splits, indicating that the gain is not split-dependent.
- The entire workflow, from data generation to performance visualization, was implemented in a MATLAB script, and the synthetic data generator is fully described.

Future work should replace the synthetic data with a comprehensive experimental database covering a wide range of pipe diameters, inclinations, and flow regimes. More sophisticated ensemble strategies (e.g., stacking with a meta-learner) and deep architectures could be explored once larger real-world datasets become available. The committee approach can also be extended to incorporate physically informed constraints, ensuring consistency with known conservation laws. The results affirm that committee neural networks, even with simple weighting, offer a pragmatic and statistically principled tool for complex multiphase flow modeling, but their practical impact must be validated on field data.

## References

- [1] Nasseh S, Mohebbi A, Sarrafi A, Taheri M. Estimation of pressure drop in venturi scrubbers based on annular two-phase flow model, artificial neural networks and genetic algorithm. *Chem Eng J.* 2008;150(1):131-8. doi:10.1016/j.cej.2008.12.011.
- [2] Alizadehdakheel A, Rahimi M, Sanjari J, Alsairafi AA. CFD and artificial neural network modeling of two-phase flow pressure drop. *Int Commun Heat Mass Transfer.* 2009;36(8):850-6. doi:10.1016/j.icheatmasstransfer.2009.05.005.
- [3] Shadloo MS, Rahmat A, Karimipour A, Wongwises S. Estimation of pressure drop of two-phase flow in horizontal long pipes using artificial neural networks. *J Energy Resour Technol.* 2020;142(11):112103. doi:10.1115/1.4047593.
- [4] Mauro A, Revellin R, Viscito L. Development and assessment of performance of artificial neural networks for prediction of frictional pressure gradients during two-phase flow. *Int J Heat Mass Transfer.* 2023;221:125106. doi:10.1016/j.ijheatmasstransfer.2023.125106.
- [5] Guesmi M, Pani SK, Othmani C, Manthey J, Unz S, Beckmann M. Estimation of pressure drop in two-phase flow using fully connected neural networks: A short communication. *Flow Meas Instrum.* 2025;106:103011. doi:10.1016/j.flow-measinst.2025.103011.
- [6] Alharthi MA, Almohammadi BA, Sharafeldin M, Abdelghany MT, Ben-doukha S, Refaey H. Heat transfer and pressure drop of Al<sub>2</sub>O<sub>3</sub>/water nanofluid in conically coiled tubes: Experimental and artificial neural network prediction. *Case Stud Therm Eng.* 2024;54:104043. doi:10.1016/j.csite.2024.104043.
- [7] Ji C, Zhang Z, Chen Z, Chen Y, Duan F. Investigations on an RBFNN-based optimization method for the two-phase flow micro-channel thermal management system of multi-chip power electronic devices. *Appl Therm Eng.* 2026;289:129936. doi:10.1016/j.applthermaleng.2026.129936.
- [8] Li Z, Lin G, Dong C, Liang H, Xu Y. Prediction of the heat transfer coefficient and frictional pressure drop for flow boiling of environmentally friendly refrigerants based on an artificial neural network. *Int J Heat Mass Transfer.* 2025;252:127428. doi:10.1016/j.ijheatmasstransfer.2025.127428.
- [9] Zhi C, Zhang Y, Zhu C, Liu Y. Numerical analysis and artificial neural network-based prediction of two-phase flow pressure drop of refrigerants in T-junction. *Int J Refrig.* 2022;137:34-42. doi:10.1016/j.ijrefrig.2022.02.005.
- [10] Almalki N. Data-driven machine learning prediction of pressure drop in two-phase flow through orifices. *Arab J Sci Eng.* 2026. doi:10.1007/s13369-026-11194-1.
- [11] Xiao J, Zhang J. Using deep learning algorithm to predict flow boiling pressure drop under ultrasound fields in mini-channels. *Int J Therm Sci.* 2025;215:110023. doi:10.1016/j.ijthermalsci.2025.110023.

- [12] Ghasemi M, Al-Safran E, Rasheed A. Flow pattern prediction of gas/liquid two-phase flow over wide range of liquid viscosity and inclination angle using machine learning. *Process Saf Environ Prot.* 2026;228:290-300. doi:10.1016/j.cherd.2026.01.034.
- [13] Liu Z, Zhang X, Liao R, He J. Gas-liquid two-phase flow pattern prediction based on the GA-PSO-DNN model and its application in energy carrier pipeline transportation. *Int J Hydrogen Energy.* 2025;202:153105. doi:10.1016/j.ijhydene.2025.153105.
- [14] Lee MS, Gwon JG, Choi HK, Park YG. Flow boiling analysis in manifold microchannels with sloped divider and artificial neural network-based flow pattern prediction. *Appl Therm Eng.* 2026;291:130261. doi:10.1016/j.applthermaleng.2026.130261.
- [15] Lu S, Chen E, Xu T, Liang X, Zhang C. Physics-informed neural network for deep learning of finned-tube evaporator performance: From the perspective of system modeling. *Int J Refrig.* 2026;185:219-32. doi:10.1016/j.ijrefrig.2026.02.021.
- [16] Nooruldeen OA, Azizifar S, Ghareeb A, Abdulkarim AH, Awad MM. Robust physics-informed neural network and swarm-based modeling of subcooled flow boiling pressure drop in metal foam tubes. *Int J Intell Comput Cybern.* 2025;18(4):676-705. doi:10.1108/ijicc-04-2025-0244.
- [17] Yang H, Wang B, Li J, Zhang G, Liu G, Zhan J, et al. A quantitative interpretation method of gas kick driven by physics-informed neural network. *Pet Sci.* 2026. doi:10.1016/j.petsci.2026.01.034.
- [18] Haghghi A, Shadloo MS, Maleki A, Jamalabadi MYA. Using committee neural network for prediction of pressure drop in two-phase microchannels. *Appl Sci.* 2020;10(15):5384. doi:10.3390/app10155384.
- [19] Xu P, Liao X, Qiu F, Cheng Z, Li W, Liu Z. Intelligent prediction of gas-liquid two-phase flow fields in jet impact negative pressure reactors: An integrated DA-WOA-CNN framework based on CFD. *Chem Eng Sci.* 2026;325:123495. doi:10.1016/j.ces.2026.123495.
- [20] Lockhart RW, Martinelli RC. Proposed correlation of data for isothermal two-phase, two-component flow in pipes. *Chem Eng Prog.* 1949;45(1):39-48.