Study of Machine Learning for Analysis of Heat Transfer Applications

Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of

Master of Engineering in Automobile Engineering

By

VIKASH KUMAR

[Examination Roll No: M4AUT24002]

[University Registration No: 163717 of 2022-2023]

Under the Guidance of

Prof. BALARAM KUNDU

DEPARTMENT OF MECHANICAL ENGINEERING

FACULTY OF ENGINEERING & TECHNOLOGY

JADAVPUR UNIVERSITY

KOLKATA – 700032

JULY 2024

FACULTY OF ENGINEERING AND TECHNOLOGY

JADAVPUR UNIVERSITY

CERTIFICATE OF APPROVAL*

This foregoing thesis is hereby approved as a credible study of an engineering subject carried out and presented satisfactorily to warrant its acceptance as a prerequisite to the degree for which it has been submitted. It is understood that by this approval, the undersigned does not endorse or approve any statement made, opinion expressed, or conclusion drawn therein but approves the thesis only for its submitted purpose.

COMMITTEE ----------

ON FINAL EXAMINATION FOR

EVALUATION OF THE THESIS ----------

***Only in case the thesis is approved**

FACULTY OF ENGINEERING AND TECHNOLOGY

JADAVPUR UNIVERSITY

CERTIFICATE OF RECOMMENDATION

I hereby recommend that the thesis presented under my supervision by ***SRI VIKASH KUMAR*** *entitled* "**Study of Machine Learning for Analysis of Heat Transfer Applications**" *be accepted in partial fulfillment of the requirements for the* ***Master of Engineering degree in Automobile Engineering***.

Thesis Supervisor

Countersigned

Head of the Department
Department of Mechanical Engineering

Dean
Faculty of Engineering and Technology

DECLARATION OF ORIGINALITY AND COMPLIANCE OF ACADEMIC ETHICS

I hereby declare that the thesis contains literature survey and original research work by the undersigned candidate, as a part of his ***MASTER OF ENGINEERING IN AUTOMOBILE ENGINEERING*** studies. All information in this document have been obtained and presented in accordance with the academic rules and ethical conduct.

I also declare that, as required by these rules of conduct, I have fully cited and referenced all the material and results that are not original to this work.

Name: **VIKASH KUMAR**

Examination Roll Number: **M4AUT24002**

Class Roll Number: **002211204002**

University Registration No: **163717 of 2022-2023**

Thesis Title: ***Study of Machine Learning for Analysis of Heat Transfer Applications***

Signature with Date:

ACKNOWLEDGEMENT

First and above all, I praise God, the Almighty for providing me this opportunity and granting me the capability to proceed successfully. This thesis appears in its current form due to the assistance and guidance of several people. I would therefore like to offer my sincere thanks to all of them.

The author expresses his deep sense of gratitude to his supervisor, Prof. Balaram Kundu, Professor, Department of Mechanical Engineering, Jadavpur University, for his inspiration, support, academic and personal guidance throughout the course work. I'm grateful to him for being very supportive in letting me pursue my interests outside of academics, and encouraging me to learn and read widely. I'm grateful for this opportunity and look forward to continue my interactions with him in future.

My thanks also go to all my batch mates who have made the atmosphere in my classes lively.

Last, but most importantly, I'm grateful to my parents, brothers and family for their love, blessings and support throughout this endeavor. This thesis, a fruit of the combined efforts of my family members, is dedicated to them as a token of love and gratitude.

Date:

(VIKASH KUMAR)

List of Symbols

$\sum$	Activation function
A_i	Area perpendicular to i'th direction
x_u^i	Boundary training data with deep neural network
U	Component of velocity in x-direction, Internal energy
V	Component of velocity in y-direction
W	Component of velocity in z-direction
Ω	Computational domain
C_1, C_2	Constants
ρ	Density
$\aleph$	Differential operator
∇	Differential or partial differential operator
x^*	Dimensionless length
T^*	Dimensionless temperature
t^*	Dimensionless time
T_∞	Environment temperature
m	Fin parameter
q_i	Heat flux in i - direction
h	Heat transfer coefficient
X_i	i'th input in the input layer of neural network
W_i	i'th weight for any neuron

dx, dy, dz	Infinite-small length
T_o	Initial temperature
t_u^i,	Initial training data
L	Length
N	Number of samples
P	Pressure
$\dot{Q}$	Rate of heat flux
$\dot{q}$	Rate of heat generation
f	Source term
x	Space coordinate in x-direction
y	Space coordinate in y-direction
z	Space coordinate in z-direction
C	Specific heat
T	Temperature
K	Thermal conductivity
α	Thermal diffusivity
t	Time

LIST OF FIGUERS

Figure 1.1: An overview of machine learning. 8
Figure 1.2: Evaluation matrices used in machine learning 11
Figure 1.3: A neural network.....11
Figure 1.4: A single neuron with weight and bias and activation function 12
Figure 1.5: Activation function (tanh)..... 13
Figure 1.6: Activation function (sigmoid)..... 13
Figure 1.7: Activation function (ReLU)..... 13
Figure 1.8: Activation function (Softmax) 13
Figure 2.1:Generalized forward model 28
Figure 2.2: Generalized inverse model 28
Figure 2.3: Generalized form of physics informed neural network. 31
Figure 2.4: Slab with specified boundary and initial condition. 34
Figure 2.5: A differential control volume. 34
Figure 2.6: Intuition of PINN used in study I. 39
Figure 2.7: Extended surface with specified boundary condition 40
Figure 2.8: A differential control volume. 41
Figure 2.9: Intuition of PINN used in study II. 45
Figure 3.1: Train and test loss vs steps (for slab). 47
Figure 3.2: A bad model: Train and test loss vs steps (for fin). 47
Figure 3.3: A good performing model, Train and test loss vs steps (for fin). 48
Figure 3.4: Distribution of temperature with space and time. 51
Figure 3.5: Temperature vs Time (at x=0.4). 53
Figure 3.6: Temperature vs Space (at t=0.5). 55
Figure 3.7: Dimensionless temperature vs Dimensionless space for mL = 4..... 59
Figure 3.8: Dimensionless temperature vs dimensionless space for mL=1. 59
Figure 3.9: Dimensionless temperature vs dimensionless space for mL=0.5. 60
Figure 3.10: Non-dimensional temperature profile for different values of mL 60

LIST OF TABLES

Table 3.1: Variation in exact and predicted temperature with space and time in slab. 49
Table 3.2: Temperature versus Time (at x = 0.4) in slab. 51
Table 3.3: Temperature vs Position (at t = 0.5). 54
Table 3.4: Predicted and exact temperature distribution at different position of x for different value of mL. 57

Table of Contents

Abstract 1
1. INTRODUCTION 2
1.1 Introduction to machine learning 4
1.2 Different types of machine learning 4
1.3 Types of supervised machine learning: 5
1.4 Types of unsupervised machine learning: 6
1.5 Algorithms used in regression 6
1.6 Evaluation metrics used in regression 8
1.7 Neural Networks 9
1.8 Components of neural network 10
1.9 Literature review 14
1.9.1 Prediction of heat transfer coefficient 14
1.9.2 Prediction of pressure drop 15
1.9.3 Optimizing heat transfer 15
1.9.4 Analysis of flow boiling/flow condensation heat transfer 17
1.9.5 Hybrid computational mode 19
1.9.6 Comparison 19
1.9.7 ML for heat transfer correlation 20
1.9.8 Contact heat transfer 22
1.9.10 Modeling Heat transfer 22
1.9.11 Study of other/miscellaneous medium/materials 23
1.9.12 Predicting critical heat flux location 25
1.9.13 Estimation of properties of nano fluids 25
2. METHOD AND PROBLEM FORMULATION 27
2.1 Physics Informed Neural Network 27
2.2 Key concept of PINN 29
2.3 Problem formulation with PINN 30

2.4 Implementation and training of a physics-informed neural network 32
2.5 Study I: Heat transfer through a slab 33
2.6 Applying physics-informed neural network to our problem 37
2.7 Study II: Thermal analysis of fin 40
2.8 Applying physics informed neural network to our study 43
3. RESULTS AND DISCUSSION 46
3.1 Results associated with study of a slab 48
3.2 Results associated with study of fin 56
4. FUTURE SCOPE AND CONCLUSION 61
4.1 Future scope associated with PINN 62
4.2 Conclusions 63
5. REFERENCES 64

Abstract

Machine learning is the outcome of the algorithmic form of statistics and a few other mathematics classes with some set of instructions, i.e., programs. This thesis explores the machine learning techniques used in the heat transfer research domain. Among the many available machine learning methods, the present work emphasizes a technique known as a Physics-informed neural network (PINN), which combines the physics associated with the problem (heat transfer) and the computational ability of a neural network. In this neural network, the loss function takes the form of a summation of different loss functions, which includes the loss in data due to PDE known as $Loss_{PDE}$, loss due to boundary condition known as $Loss_{BC}$, and loss due to initial condition known as $Loss_{IC}$. In this work, the PINN technique is used to study heat transfer behavior through a slab and a fin under specified boundary and initial conditions, two of the most common problems in the thermal domain. Solutions obtained from this method are validated using analytical solutions.

Chapter-1

1. INTRODUCTION

From the knowledge of thermodynamics, we understand energy conversion from one form to a subsequent form. At the same time, science associated with heat transfer is concerned with the study of the rate of heat transfer within a system. Such heat transport always occurs in a temperature gradient, i.e., from higher to lower temperatures. Heat transfer science focuses on finding the heat flow characteristics and temperature gradient in many practical scenarios like heat exchanger design, nuclear reactor core, aerospace industries, and many more research domains. Most real-world problems include all modes of heat transfer, such as conduction, convection, and radiation, simultaneously. However, to simplify our analysis, we only consider the dominating mode of heat transfer [1]. Some of the crucial problems in this domain are thermal resistance and insulation [2], study of heat loss [3], thermal stresses [4], heat exchanger design [5], temperature control [6], cooling efficiency [7], phase change management [8] and environmental impact [9]. There are several methods available for the study of heat transfer problems. Some of the most popular methods in research works are:

Analytical methods: These methods solve the governing differential equation of heat transfer using mathematical techniques. They include methods of variable separation, integral transforms, and similarity solutions. They provide an exact solution to the problem. However, they have limited applicability due to the presence of assumptions and the complex nature of the governing equation [1,10].

Numerical methods: for complex geometries and boundary conditions where traditional mathematics is not feasible. It discretizes the governing equation and uses iterations to solve them. Some standard numerical methods are finite difference method (FDM), finite element method (FEM), finite volume method (FVM). This method comes with the intensive computational cost and truncation error associated with equation [11,12,13].

Experimental methods: This method involves experiments to measure parameters associated with heat transfer [14,15]. Cost, scalability, measurement error, and time consumption are some of the most challenging aspects of this method.

Dimensional analysis and similarity: This method uses dimensionless numbers to define the heat transfer problem and its influencing parameters [16,17]. It provides an approximate solution and cannot determine the complexities of an actual system. It is also limited to a specific number of problems where dimensionless numbers can be used.

Empirical correlations: This relationship derived from experimental data and approximated solutions is used for heat transfer problems. This method is mainly used in convective heat transfer for various configurations [18].

Network methods: Heat transfer problems are tackled using an electoral analogy where electrical components represent thermal capacitance, resistance, and heat source/ sinks [19,20]. This method is unsuitable where thermal resistance and capacitance aren't clearly defined.

Considering the limitations associated with the methods mentioned above, machine learning can fix many of the issues related to these methods. It can help in the following ways:

Analytical method: Implementing ML to the problem can tackle complex geometry and boundary conditions by learning from large datasets. That overcomes the limitations of traditional mathematical techniques [21].

Estimated computational costs with the numerical method can be potentially reduced using ML models, which can help overcome errors with a numerical technique [22].

Cost labelled with the experimental data collection can be reduced by predicting the outcome based on a smaller set of experimental data [23]. ML can model complex non-linear relationships between variables that are difficult to capture with traditional dimensional analysis [24].

Machine learning is a theoretical concept and a practical tool for creating hybrid and surrogate models. These models can be seamlessly integrated with CFD simulations, effectively reducing the computational resources required for repeated simulations [25].

It can develop more accurate correlations by learning from large datasets and capturing complex relationships between variables [26]. ML enhances network methods by capturing complex and more detailed interactions within thermal systems [27].

1.1 Introduction to machine learning

It is a tool combined with statistics and computer codes (i.e., a set of instructions) used for data-driven problems [28]. Statistics, a part of mathematics, explains the hidden relation within the data. Meanwhile, the computer code translates some form of thing (data) into some other form (bit), which utilizes the hash power of the computer to compute it. The machine learning field grew from old statistics and artificial intelligence. Machine learning algorithms are used to gather an understanding of collected data, clarify the interpretation of phenomena in the form of models, predict/forecast the future values of phenomena, and find strange behavior [29] exhibited by a phenomenon. But the question is, when should we deploy machine learning rather than traditional programming?

Machine learning tools are susceptible to input data. These tools perform well even if there is a lot of variation in input data. They can train themselves to perform, which is outside the limit of direct programs. Applying machine learning to such problems includes programs that decode handwritten text, where a fixed program can adapt to variations between the handwriting of different users; spam detection programs, which adapt automatically to changes in the nature of spam e-mails; and speech recognition programs.

1.2 Different types of machine learning

Machine learning has branched into several subfields (as shown in Fig. 1.1) dealing with different learning tasks. Some of them are:

a. Supervised machine leaning: Labelled training data in supervised learning algorithms learns the relationship between inputs and outputs [30]. The data used in supervised learning is labelled, which contains examples of inputs (called features) and correct outputs (labels). The algorithms analyse a large dataset of training pairs to find a desired output value when asked to make a prediction on new data. When the model has been trained and tested, it can be used to make predictions on unknown datasets based on the previous training. Supervised machine learning algorithms make it easier for researchers/organizations to create complex models that make better predictions. So, they can be widely used across various industries/fields, including research [31], marketing [32], financial services [33], healthcare [34], and many others.

b. Unsupervised machine learning: Unsupervised learning (or knowledge discovery) algorithms learn from data without human supervision. Unlike supervised learning, unsupervised machine learning models deal with unlabelled data and are allowed to find patterns and information without any guidance/instruction [35]. Algorithms learn without any kind of labels or any prior training. The model is given raw and unlabelled data and has to interpret its own rules and relation for the information based on differences, similarities, and patterns without any instructions on how to work with every single piece of data.
These algorithms are well suited for more complex processing tasks, such as arranging large datasets into clusters. They are helpful in identifying previously undetected patterns in data and can also identify valuable features for categorizing data [36].

c. Reinforcement learning: Reinforcement learning (RL) learns to optimize sequential decisions, which are taken recurrently across time steps. Sequential decision means a situation where the decision maker, in RL terminology known as an agent, makes successive observations of the process before making a final decision [37]. It tries to mimic how humans learn. Humans can understand complex and different tasks like swimming, gymnastics, or connecting an instance with another instance [38]. More specifically, in practical use cases of RL, it tries to acquire the best strategy for making repeated sequential decisions across time (i.e., dynamic state) under uncertainty [39]. It does so by interacting with a simulator of the stochastic dynamic system of interest, also called an environment, to learn such winning strategies [40]. A strategy to take repeated sequential decisions across time in a dynamic system is also known as a policy. RL tries to learn the winning policy of choosing actions in different states for a dynamic system [41].

1.3 Types of supervised machine learning:

a. **Classification**: In Classification, datasets are used to group particular data by predicting a label (categorical) or output variable based on provided input data. Classification is used when output variables are categorical (i.e., in the form of yes-no, true-false, etc.), meaning there are two (Binary Classification) or more than two classes [42].
b. **Regression**: Regression is a statistical approach that relates the relationship between a dependent variable (target variable) and one or more independent variables (predictor variables) [43]. The objective is

to establish the most suitable mapping (function) that defines the connection between these variables. It approaches finding a best-fitting model that is utilized to make predictions or draw conclusions. The regression analysis problem works if the output variable is a real/continuous value [44], such as "temperature" or "geometric dimension."

1.4 Types of unsupervised machine learning:

a. **Clustering**: Clustering techniques explore raw and unlabeled data by breaking it down into small chunks (or clusters) based on similarities (or differences) [45]. They are used in a variety of applications, including customer segmentation [46], fraud detection [47], and image analysis [48]. Clustering algorithms split data into natural groups by finding similar structures or patterns in uncategorized data.
b. **Dimensionality reduction:** The dimensionality reduction technique reduces the number of features/dimensions in a dataset. Extensive data is generally suitable for ML, but it is more challenging to visualize and draw conclusions from it [49]. This technique extracts essential features from the dataset that reduce the presence of irrelevant features. It uses principal component analysis (or PCA) [50] and singular value decomposition (or SVD) [51] algorithms to reduce the number of data inputs without compromising the integrity of the properties in the original data.

1.5 Algorithms used in regression

a. **Linear regression:** Linear regression is one of the simplest and most widely used statistical tools. It assumes that there is a linear relationship between the independent variable and dependent variables [52]. This means that the change in the dependent variable is proportional to the change in the independent variables.
b. **Polynomial regression:** Polynomial regression models a non-linear relationship between the dependent and the independent variable. It adds polynomial terms to the linear regression model to capture more complex relationships [53].
c. **Support vector regression:** The support vector regression (SVR) algorithm is based on the support vector machine (SVM) algorithm. SVM is a type of algorithm used for classification problems, but it can also be used for regression problems. SVR works by finding a

hyperplane [54] (a plane that differentiates between two classes) that minimizes the sum of the squared residuals between the predicted and actual values [55].

d. **Decision tree:** A decision tree regression algorithm builds a decision tree that predicts the target value, whereas a decision tree is a tree-like model that contains nodes and branches. Every single node represents a decision, and every single branch represents the outcome of that decision [56]. Decision tree regression aims to build a tree model that can accurately predict the target value for new data points [57].
e. **Random forest:** Random Forest regression is an ensemble method. Multiple decision trees are combined to predict the target value [58]. Random forest regression works by building a large number of decision trees, each trained on a different subset of the training data. The final prediction is made by averaging all the trees' predictions [59].
f. **Regularized linear regression**
    a. **Lasso regression:** Lasso regression is one of the types of linear regression, also known as L1 (in statistics, it is known as the sum of absolute values). Regularization is used to prevent overfitting. For this, it adds a penalty term to the loss function/error function (quantifying how well the model makes predictions) that forces the model to use some weights and sets others to zero [60]. This method can also be used to find the minimum number/threshold features by customizing the regularization parameter [61].
    b. **Ridge regression:** Ridge regression/L2 regularization is a regression prediction algorithm that prevents overfitting by decreasing the weight of abnormal data. Overfitting occurs when the model learns the training data too well but cannot generalize to new data [62].

$$L_2\ regularization\ term\ = \|w\|_2^2 = w_1^2 + w_2^2 + w_3^3 + \ldots\ldots w_n^2$$

Where *w* represents the assigned weight, weights close to zero have little effect on the model, while outlier weights have a huge effect.

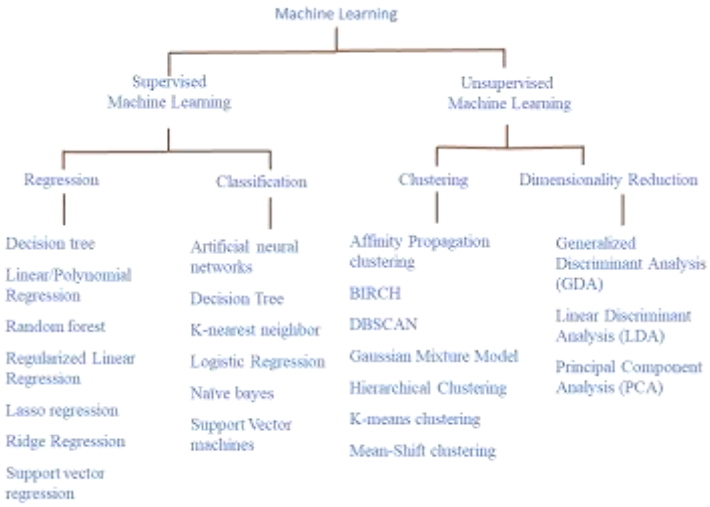

Figure 1.1: An overview of machine learning.

1.6 Evaluation metrics used in regression

a. Mean absolute error: It expresses the mistakes made by the model, known as errors. It is defined as the average/mean sum of all such errors [63].

$$MAE = \sum_{Y=1}^{N} \frac{|Predicted\ output_Y - Actual\ output_Y|}{N} \quad (1.1)$$

b. Mean square error: It expresses the squared distance within actual and predicted values. Squaring error is performed to avoid negative terms, and it is the benefit of MSE [64]. One of the major advantages of using MSE is that its graph is differentiable, so it can easily be used as a loss function/objective function [65].

$$MSE = \sum_{Y=1}^{N} \frac{(Predicted\ output_Y - Actual\ output_Y)^2}{N} \quad (1.2)$$

c. R2 Score: R square (or R2 score) expresses the model's performance. It determines the relative improvement of a regression line over a mean line. R2 squared is sometimes called the goodness of fit or the coefficient of determination [66].

$$R2\ Squared\ = 1 - \frac{Squared\_sum\ error\ of\ regression\ line}{Squared\_sum\ error\ of\ mean\ line} \quad (1.3)$$

d. Root mean square logarithmic error: This evaluation adds matric one to both (i.e., predicted and actual output) terms before operating with the natural log function. This method is preferred when the data set contains outliers and many zeros. A lower value of RMSLE indicates a small error, i.e., better prediction performance [67].

$$RMSE\ = \sqrt{\frac{\sum_{Y=1}^{N}(log(Predicted\ output_Y+1)-(log(Actual\ output_Y+1))^2}{N}} \quad (1.4)$$

e. Root mean squared error: It is square root of mean squared error. It makes the interpretation easy because it gives output value in the same unit as the required output variables have. It is mostly used with deep learning techniques [68]**.**

$$RMSE\ = \sqrt{\frac{\sum_{Y=1}^{N}(Predicted\ output_Y-Actual\ output_Y)^2}{N}} \quad (1.5)$$

Few other evaluation metrices are also shown in Figure. 1.2.

1.7 Neural Networks

Neural networks are the fusion of artificial intelligence and brain-inspired design that reshapes modern computing. With intricate layers of interconnected artificial neurons, these networks emulate the intricate workings of the human/animal brain. Neural networks can adapt to changing input to generate the best possible result without redesigning the output criteria. A neural network contains layers of interconnected nodes. Each neuron has two adjustable parameters, weight W and bias. Different types of neural networks, from feedforward to recurrent and convolutional, are customized for specific tasks [69]. Figure 1.3 illustrate the general procedure of a neural network.

1.8 Components of neural network

a. **Input layer:** It's the layer in which input is provided to the model. The number of neurons in this layer equals the total number of features in the data.

b. **Hidden layer:** The output from the Input layer is then fed into the hidden layer. Depending on the model and data size, there can be many hidden layers. Each hidden layer can have different numbers of neurons, which are generally greater than the number of features.

c. Output layer: The output from the hidden layer is then fed into a logistic function like sigmoid or softmax, which converts the output of each class into the probability score of each class.

d. **Activation function:** A neural network without an activation function acts as a linear regression model. Assigning an activation function (Fig. 1.4) to neurons introduces non-linearity to the neural network. Some popular activation functions are, sigmoid [70], tanh, ReLU, and softmax [71], plotted below in Figs. 1.5-1.8.

$tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$	$sigmoid = \frac{1}{1 + e^{-x}}$	$ReLU = max\,(0, x)$ $= \begin{cases} x, & x > 0 \\ 0, & otherwise \end{cases}$

e. **Initialization:** Initialization in a neural network simply means assigning the initial value of weight and bias for neurons. Different weight initialization methods are used, such as normal initialization, constant initialization, Lecum initialization, random initialization, Xavier initialization (also known as Glorot normal), and He initialization [72].

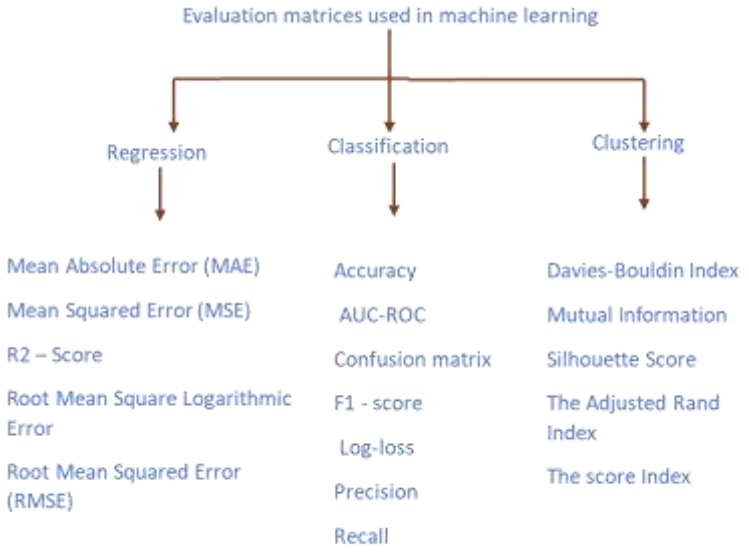

Figure 1.2: Evaluation matrices used in machine learning

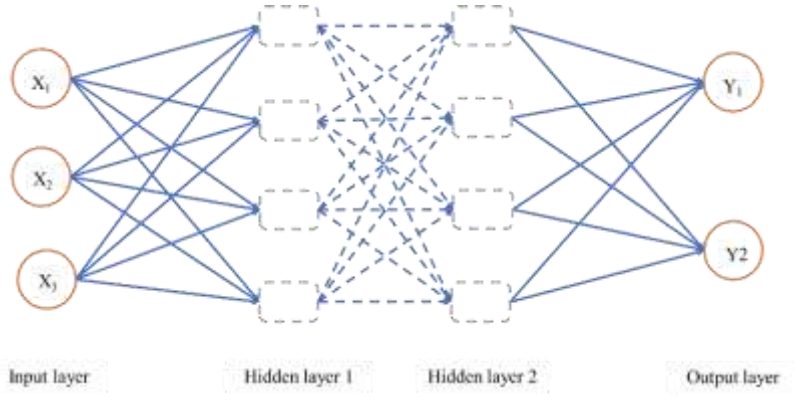

Figure 1.3: A neural network

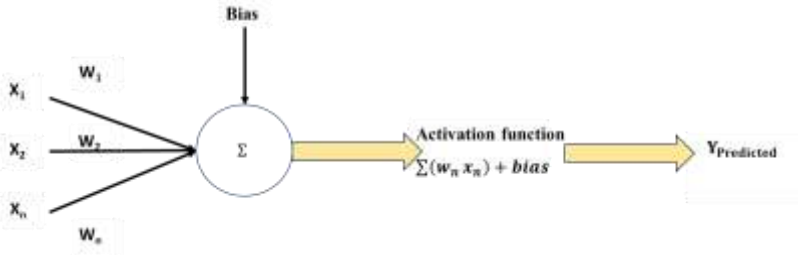

Figure 1.4: A single neuron with weight and bias and activation function

f. **Optimization:** It is a process of selecting the best solution from a set of available solutions. It deals with either maximizing or minimizing the objective function or loss function. Some of the most commonly used optimization techniques are stochastic gradient descent, Adam, RMSprop, random search, etc. [73]. One of the most important terminologies used in optimization is learning rate. The learning rate is nothing but the step size at each iteration while trying to minimize a loss function.

g. **Metrics:** Evaluation matrices are used to evaluate the performance of the machine learning model. Each model targets generalizing well on new or unseen data, and evaluation metrics help determine how well the model generalizes on an unseen data set.

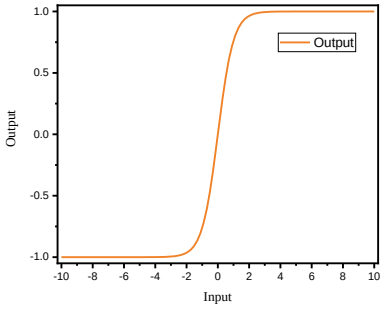

Figure 1.5: Activation function (tanh)

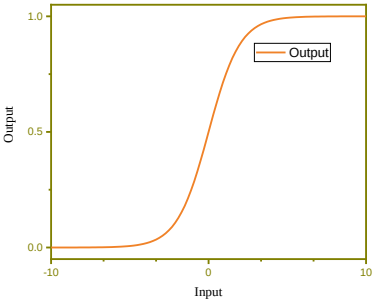

Figure 1.6: Activation function (sigmoid)

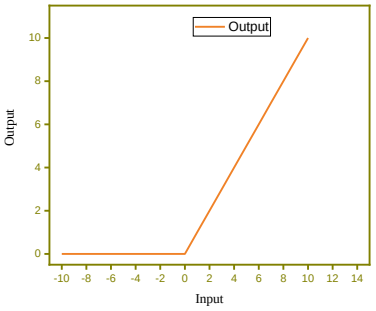

Figure 1.7: Activation function (ReLU)

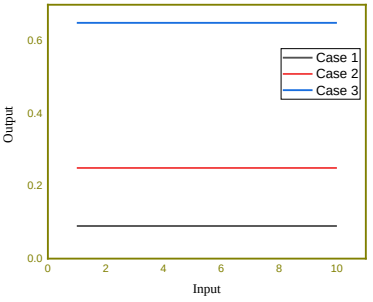

Figure 1.8: Activation function (Softmax)

1.9 Literature review

1.9.1 Prediction of heat transfer coefficient

A temperature and a velocity gradient are observed whenever a flow occurs on any surface due to boundary development. Based on the nature of flow, convective heat-transfer problems are either natural convective heat-transfer or forced convective heat-transfer. Both of these types divide the boundary layer zone into either a laminar region or a turbulent region. Newton's law of cooling $(i.e. Q = hA\Delta T)$ is only a defining formula for the heat transfer coefficient (h) and does not relate the heat transfer coefficient and the factor that impacts it. Some of the most common methods of predicting the heat transfer coefficient are:

(a) **Analogy method:** The analogy method measures one of two distinct physical occurrences to determine the fundamental relationship between them based on the similarity of their governing equations [74].

(b) **Experimental method:** This method is also known as the correlation technique. It is based on experiments and a trial-and-error approach. Dimensional analysis is also used with this technique [75].

(c) **Numerical Method:** In these techniques, numerical methods are used to solve the energy and Navier-Stokes equations under prescribed boundary conditions [76].

Apart from the three methods mentioned above, a direct data-driven approach is also used to predict the heat transfer coefficient. An ML-based approach by Acikgoz et al. [77] for the prediction of the heat-transfer coefficient of the radiant wall with a cooling system and mixed forced convection for these two neural networks with different input parameters (temperature, velocity, and heat transfer rate) are assigned. In another problem, Acikgoz et al. [78] did experiments for the data generation, and these data were used in the development of ANN using MATLAB and studied the heat transfer coefficient. Verma et al. [79] have suggested an ANN model to estimate the HTC of the radiant heating and cooling system. An experimental cum computational study of the radiant wall with a cooling effect is explained by Çolak et al. [80], and they have used ANN for the estimation of convective and radiative heat flux. Estimating the heat transfer coefficient of multiple impinging jets on a hot steel surface

(quenching) with the consideration of jet velocity, surface moving speed, and temperature makes the problem difficult. To find the value of HTC, Xie et al. [81] used experimental data and an ML model and used its value to calculate the transient heat flux in two dimensions.

1.9.2 Prediction of pressure drop

A change in the overall pressure of fluid-carrying equipment (e.g., heat exchanger, tubes, etc.) between two points is known as a pressure drop. Pressure drop in the heat exchanger is directly related to the velocity of the fluid. The higher the velocity, the higher the heat transfer coefficient, and the higher the pressure drops. The optimum heat exchanger design demands a higher value of heat transfer coefficients but a lower pressure drop value. This is the main reason a researcher must find a way to predict the pressure drop. Alejandro et al. [82] used ANN and data handling methods to develop empirical models calculating the pressure drop. Bhattacharyya et al. [83] presented a statistical tool and created an ANN for a hybrid tape (combination of wavy tape and grooved spring tape) type circular channel and estimated the pressure drop and heat transfer coefficient. Using two ensemble boosting algorithms and one bagging algorithm, a double pipe type heat exchanger analysis was done by Sammil et al. [84], and the thermohydraulic parameters were predicted. A similar type of pipe with a coiled-wire turbulator is studied by Celik et al. [85] with four regression models: Support vector regression, Gaussian process regression, Multilayer perceptron network, and Random Forest. Multi-linear regression is applied to compare results, and the drop in pressure is estimated. For the study of condensation pressure drop and coefficient of heat transfer in a horizontal macro and microchannel, Hughes et al. [86] used a combination of flow-regime-based relation and model-based ML regression that used random forest, ANN, and support vector regression.

1.9.3 Optimizing heat transfer

Analysis of heat transfer in solid-fluid interaction known as Conjugate heat transfer. This kind of analysis is required to optimize heat exchangers, design turbine airfoils, improve the heat transfer characteristic in electronic components, etc. For this, the most common approaches are model-based analysis and model-free analysis. The model-based approach is problem-dependent and requires domain knowledge, and the model-free analysis and model-free approach do not require domain knowledge. It uses the input-output database to optimize the desired output and estimates the

input to get the desired output. Dutta et al. [87] proposed a non-linear optimization replacement strategy for the optimization of the airfoil shape of the turbine vane; for this linear regression, the ML model is used, which optimizes the problem in terms of design, cost, and life of the component, and flow rate of coolant. Micro fins are optimized to enhance heat transfer capabilities. Larra˜naga et al. [88] used the performance evaluation criterion as the ratio of geometry's thermal and hydraulic performance parameters as an objective function for the data-driven model. Kang et al. [89] introduce three ML algorithms, ANN, support vector regression, and random forest, and predict the average: heat transfer and net energy loss. Balachandar et al. [90] suggested using the ANN-GA-based technique to optimize fin performance. ANN is used to estimate base temperature, and ANN is also trained using a limited data set. Then, a genetic algorithm is introduced to the trained ANN to find the optimized geometry. Integration of ML, GA, and CFD by Wang et al. [91] estimated the influence of the geometry of the finned heat-pipe radiator. It gave the optimal geometry that improved the heat transfer. Shi et al. [92] suggested using a surrogate model in combination with ML technology. This approach achieved an optimal dimple arrangement that optimized the heat transfer. Dadhich et al. [93] generated a dataset using experiments on annulus tubes filled with water-based nanofluids. They used these data to develop the ANN model, and these ANNs are in better agreement with experimental results.

Design parameters or processes can also be optimized using ML and its algorithms. Processes involved in the design of thermal devices are complex. Wen et al. [94] used ML in combination with ANN and genetic algorithms to solve this issue. Yu et al. [95] suggested a significant way to save energy and reduce emissions by improving the design of heat transfer equipment by implementing the ML technique to update the surrogate model. Nusselt number predictor model uses volumetric concentration of nanofluids, winding number, and Prandtl number for helically coiled heat exchanger as input parameters suggested by Baghban et al. [96] consist of a multi-layer perceptron ANN, ANFIS (adaptive neuro-fuzzy inference system) and LSSVM (least-square support vector machine). Estimation of the heat transfer coefficient during condensation of steam in the presence of air outside a heated tube studied by Cao et al. [97] used experimental data to build a back-propagation neural network model. Factors that influence condensation heat transfer (e.g., pressure, subcooling, tube diameter, air mass fraction) are used as input parameters, and the coefficient of heat transfer is used as the output of the model. A method suggested by

Elboughdiri et al. [98] was used to increase the Nusselt number and reduce the pressure drop i.e., optimized the disk-shape micro-channel heat sink with the feature of a variable number of spiral micro-channel, based on ANN, which is developed as forecasting model that takes several channels and volume flow rate as input and produce Nusselt number and pressure-drop as output. Using the ML learning regression model, the Performance of fins for axially finned tube heat exchangers is investigated by Krishnayatra et al. [99], who studied the variational effect of fin spacing, fin material, fin thickness, and coefficient of convective heat transfer on total effectiveness and overall efficiency and estimated the thermal Performance of fin—for the study of heat exchangers with corrugated (i.e., variation in pitch and depth) and non-corrugated pipes by, Verma et al. [100] developed an ANN that predicted the Reynolds number, coefficient of heat transfer and Nusselt number. For BPHE (i.e., brazed plate heat exchanger), Longo et al. [101] used ANN to predict the coefficient of boiling heat transfer by considering the refrigerant characteristics, plate shape, size, and operating conditions. Friction factor and dimensionless factors associated with heat transfer are the crucial parameters for the optimization of Plate-finned heat exchanger (PFHE) Performance. The approach followed by Kedam et al. [102] uses ANN (more specifically, an MLP). It predicts the parameter mentioned above for various fin types (including wavy fins and off-set strip fins), which utilizes the Bayesian regularization learning technique. Two ANNs were developed by Çolak et al. [103], taking coil and tube diameter, Reynolds number, curvature ratio, and mass flow rate as input parameters and forecasted the value of Nusselt number, pressure drop and heat transfer coefficient for shell-and helically-coiled tube heat-exchanger. Moradkhani et al. [104] developed three ML algorithms (Gaussian process regression, multi-layer perceptron, and Radial Basis Functions) for smooth helical-coiled tubes and modeled the boiling heat transfer coefficient.

1.9.4 Analysis of flow boiling/flow condensation heat transfer

International space agencies are investigating using two-phase heat management systems to sustain astronauts living aboard future spacecraft and planetary colonies. The primary driving forces behind these endeavors are the augmentation of power consumption efficiency and the reduction of total weight and volume; these characteristics are direct outcomes of the heat transfer enhancement attained by flow boiling and condensation [105]. Buoyancy, which is proportional to the product of gravity and the density

difference between liquid and vapor, reflects the effect of gravity on flow boils. Due to the significant density difference, buoyancy can significantly impact how vapor moves in relation to liquid and, in turn, how well heat transfer occurs. The focus of microgravity research has recently been urged to be shifted from pool boiling to flow boiling to attain the high CHF values and heat transfer coefficients needed for space applications, such as space suits, space vehicles, Earth-orbiting stations, satellites, fighter aircraft, and so forth [106] [107]. Instead of traditional empirical correlations, ML shows impressive potential for predicting two-phase flow parameters. Tarabkhahet al. [108] described four different ML models: multi-layer perceptron ANN, Support vector regression, extreme gradient boosting, and k nearest neighbors, and predicted the heat transfer coefficients and pressure drop. Guangya Zhu et al. [109] compared previous conventional and current ML-based flow boiling investigations to get highly efficient aircraft thermal management. Computational simulation and traditional correlation are slow and less accurate in predicting heat transfer. Qiu et al. [110] forecasted the heat transfer coefficient in mini-micro channel/heatsinks for flow boiling using ANN and physics-based data-driven techniques. A dataset can be prepared with the extraction of information from images, and an ML model known as a Convolution Neural Network (CNN) can be applied to a specific problem. Junior et al. [111] prepared a dataset and applied a CNN model to study condensing refrigerant in a vertical straight tube. Data generated from experimental setup Zhu et al. [112] used an ANN and predicted heat transfer performance for flow boiling and condensation.

A data-driven ML model (non-linear regression) was proposed by He et al. [113]and predicted the flow coefficient. Four ML models, ANN, Random Forest, Ada boost, and Extreme gradient boosting, were developed by Liwei Zhou et al. [114] to predict the heat transfer coefficient during flow condensation.ML technique developed by Tang et al. [115] consists of nine algorithmic models predicting the Nusselt number for the bubble condensation. A high-fidelity ML framework suggested by Khodakarami et al. [116] predicted the heat transfer coefficient for condensation. ANN, explained by Bard et al. [117], predicts fluid characteristics for nucleation or bubble growth in flow boiling. The approach followed by Yang et al. [118] uses feature extraction and classification algorithms to determine the characteristics of different flow patterns. To improve HVAC systems with two-phase flow, Al-Jarrah et al. [119] estimated the phenomena of heat transfer during the condensation of CO2in porous medium by combining

two neural-network models and two adaptive neural-fuzzy inference systems with the prediction of internal heat transfer coefficient.

1.9.5 Hybrid computational mode

The hybrid computational method combines AI and mathematical models (analytical or numerical) and applies them to solve a wide range of problems in the field of engineering or research. Such a model using CFD (finite volume approach) and AI-based models is developed by Zhao et al. [120] to study temperature distribution in nanofluid. Deb et al. [121] gathered the data using CFD analysis and used these data to generate an ML model that predicted the mean final temperature of the water and the drop in the pressure. Manshadi et al. [122] used a deep neural network with Long short-term memory in this technique to compare the different methods and predict the effect of radiative and conductive heat transfer in polymethylmethacrylate (a type of polymer used in various kinds of actuators and sensors). Data from 5,000 CFD simulations for turbulent flow in pipes used with ANN Koroleva et al. [123] analyzed the heat transfer, thermal-hydraulic performance, and pressure drop. Babu et al. [124] used a peri-dynamics-based multivariate linear regression model to analyze thermal behavior. Naphon et al. [125] used a combination of CFD and ANN to study the behavior of pressure drop and heat transfer due to the influence of jet impingement in micro-channel heat sink. ANN model utilizes the LMB (i.e., Levenberg-Marquardt Backward-propagation) algorithm for training. For the analyzing one of the complex problems, including entropy generation, mixed convection, and non-linear thermal radiation with local thermal non-equilibrium within the porous medium (filled with nanofluid), Alizadeha et al. [126], a hybrid technique, collected data from CFD analysis and used these data to train an ML model. Support vector regression approximates the Nusselt number, Shear stress, velocity, temperature, entropy generation, and the Bejan number function.

1.9.6 Comparison

Under comparable working conditions, radiators fitted with brazed, double-U grooved, and standard pipes are compared. The analytical Number of the Transfer Unit Method is used to validate the experimental evaluation of the radiators' heat transfer characteristics. Using the acquired experimental data, an Artificial Neural Network model has been created to ascertain the heat transfer rate for every radiator. The numerical model's data have been thoroughly compared to the practical model. The data

extracted from the ANN structure and the goal data have been compared to analyze the model's performance analysis thoroughly. Kuzmenkov et al. [127] simulate the improved heat transfer properties of turbulent airflow in a circular tube with transverse ribs that are artificially rough. Kuzmenkov examined three alternative modeling techniques. The first dataset is formed via several CFD simulations, and the second dataset is taken from several classical works. A deep feed-forward neural network is created to forecast Nu and friction factors for rib roughness and flow parameters. The first and second datasets are used separately to train the ANN and a combination of datasets that consistently produce high-quality predictions. A comparison of all results with experimental data and CFD modelled values displays the best outcomes of the experiment and ANN technique.

Significant uncertainties exist in the heat transport correlations, and traditional tests and CFD simulations need a lot of time and processing power. Zheng et al. [128] set out to develop a dependable technique for more rapid and precise estimating the Heat Transfer Coefficient of heat exchange channels. The General Regression Neural Network and Random Forests models forecast the heat-exchange performances of channels with varying height bulges. These models are trained using hundreds of CFD simulation data. The total height of the bulge is likewise correlated with HTC. Following a specific reduction in the overall bulge height, HTC's progress becomes restricted. The findings demonstrated that the HTC of channels with various bulge configurations can be accurately predicted.

1.9.7 ML for heat transfer correlation

A standard method for estimating the heat flux between a fluid and its surface is to use heat transfer correlations. The main determinants of these formulae are geometry, working fluid, and operating circumstances. Analytical solutions can be used to calculate the features of laminar flow under certain assumptions. However, to estimate heat transfer for turbulent flows, one needs to use numerical models or experimental data [129]. Nie et al. [130] used machine learning techniques to forecast the HTCs for horizontal tube flow condensation. To assess five ML models, a comprehensive database encompassing a wide variety of fluids and experimental settings is put together. To ascertain the heat transfer properties, a new universal correlation is created based on the parametric importance analysis carried out by the XGBoost models. Five machine learning models were based on the algorithms of Extreme gradient boosting

(XGBoost), random forest (RF), convolutional neural network (CNN), artificial neural network (ANN), and K-nearest neighbors (KNN).

To derive a fundamental correlation for predicting convective heat transfer of the nanofluids loaded with graphene, Savari et al. [131] examined the heat transfer performance of the Water/ethylene glycol-based graphene nanofluid and the Water/ethylene glycol-based graphene nanofluid nitrogen-doped graphene. The impacts of significant parameters on heat transfer and fluid flow characteristics of an automobile radiator are modeled using an ANN and adaptive neuro-fuzzy inference system (ANFIS), and the modeled data is then compared with experimental results for testing. Following modeling, the effective model was chosen by comparing the outcomes of the ANFIS and BPNN models. In this manner, two new nanofluids were created initially. Then an effective model was used to simulate the Nusselt number in relation to changes in the inlet temperature, Reynolds number (Re), and Prandtl number (Pr).

Absorbers and desorbers based on microchannel membranes are essential parts of small and effective absorption refrigeration systems. More sophisticated correlation models are desperately needed to increase the accuracy of existing empirical correlations for predicting the properties of solution pressure drop and heat/mass transfer. According to Zhai et al. [132], new models for the Nusselt number (Nu), Sherwood number, and friction factor of microchannel membrane-based desorber and absorber, respectively, are developed using three machine learning algorithms: Random Forest (RF), Least-Squares Support Vector Machine (LS-SVM), and Genetic Algorithm-optimized Back Propagation Artificial Neural Network (GABPNN), based on experimental results. With machine learning support, these models effectively increase the prediction accuracy for both the absorber and the desorber.

Steam condensation is a significant phenomenon in nuclear reactors during severe accidents. A biologically inspired machine learning technique called multigene genetic programming (MGGP) was examined by Tang et al. [133] and used to create novel correlations for condensation HTC of a steam-non-condensable gas mixture over a vertical tube in the turbulent free convection regime. It illustrates how MGGP has promise in making precise, concise, and clear models for multiphase flow processes like steam condensation in the presence of non-condensable gas and complex heat transfer. Metals, metal oxides, and polymers are among the components that are typically used to create nanoparticles. It is essential to

assess the heat transfer the nanofluids achieve to develop energy systems that help.

Most models created to explain fluid heat transfer necessitate calculating the Nusselt number, which measures the proportion of conductive to convective heat transfer in a particular system. Guzman-Urbina et al. [134] estimated the local Nusselt number of nanofluids by deriving a correlation using Genetic Programming (GP). Using an evolutionary algorithm, the technique creates correlation equations for examining the relationship between the flow regime, flow characteristics, system setup, and nanoparticle properties. Developing multi-variable heat transfer correlations requires much less work, thanks to machine learning, which may also quickly expand the parameter domain.

For a high-order nonlinear heat transfer problem, Kwona et al. [135] employed the random forests algorithm to forecast the convection heat transfer coefficient. Next, the RF regression's interpolation capacity is evaluated.

1.9.8 Contact heat transfer

Analysis of heat transfer at the interface between two contacting bodies is one of the complex problems of heat transfer. Vu et al. [136] introduced an ML model based on supervised learning that predicted the interaction of heat across the interface between two bodies.

1.9.10 Modeling Heat transfer

For laminar and turbulent flow with magnetic nanofluids flowing inside a pipe, Zhang et al. [137] calculated the convective coefficient of heat transfer using ML technique by selecting different machine learning approaches viz multiple linear regression, least square-support vector mechanism, and radial basis function-backpropagation were used. A method for adjusting supercritical pressure fluid turbulence models using high-fidelity simulation (DNS) data is presented by Cao et al. [138]. An iterative DNS-DNN-RANS framework is suggested to create explicit closures for turbulent momentum diffusion and turbulent thermal diffusion of turbulent heat transfer at supercritical pressure. For a passive-containment cooling system in a light water reactor, a model that predicts condensation heat transfer suggested by Lee et al. [139] developed a correlation for the coefficient of heat transfer that uses the ML method. Data was used to train and test the ML model; a pseudo dataset was created using the existing condensation model, and a Neural network model with a

convolution neural network was used. Developing frictional pressure gradient correlations and heat transfer for the condensation of binary mixtures can be computationally demanding and intricate. Because of mass transfer and temperature glide, binary mixes add another layer of complication. Hughes et al. [140] created a variety of dimensionless parameter inputs known to impact pressure drop and heat transfer, which were used to develop predictive machine learning models that are accurate but computationally inexpensive. The Nusselt number was discovered to be predictable by the SVR model. The GB algorithm predicted frictional pressure drop. Mudhsh et al. [141] used cutting-edge machine-learning algorithms to model the thermo-hydraulic behavior of a spiral plate heat exchanger. Different flow channel pitches and cross-sectional sizes are considered when predicting the output temperatures of hot and cold fluids. The suggested model aids in the estimation of the thermal performance and attributes by heat exchanger manufacturers. To forecast the outlet temperatures of the working fluids of helical plate heat exchangers, an improved random vector functional link (RVFL) optimized fire hawk optimizer is intended to be provided by the developed model.

1.9.11 Study of other/miscellaneous medium/materials

An experimental study of the melting of n-octadecane (a phase change material) in combination with ANN developed by Motahar [142] estimates the melting parameter of PCM. An MLP feed-forward NN is trained using the Levenberg-Marquardt algorithm (uses experimental data) and predicts the melted volume fraction and Nusselt number as output by taking Rayleigh, Stefan, and Fourier numbers as input parameters.

The thermal diffusivity of volcanic rock studied by Khan et al. [143] uses a genetic algorithm. Ensembled learning tree, Gaussian process regression, Support vector machine, and decision tree ML model are integrated with genetic algorithm and forecasted thermal diffusivity. Transfer of heat and storage of energy of ground (i.e., soil) is directly related to thermal diffusivity. Direct measurement is challenging due to the limitations of sensors and variable physical characteristics of soil, viz. mineralogy, texture, moisture content, bulk density, etc. Hence, Li et al. [144] suggested a method that comprises ML models like KNN, Random Forest, MPL, SVR, decision tree, and GBDT (gradient boosting decision tree). Accurate prediction of deterioration of transfer of heat in supercritical fluids (state of fluid beyond the critical temperature and pressure) is influenced by many parameters, which makes traditional correlations

challenging to predict. Zhang et al. [145] developed ANN and predicted the Nusselt number. Dongliang et al. [146] studied water in a supercritical state and predicted the coefficient of heat transfer by altering an ML model's regularization penalty parameter, slack variable epsilon kernel function parameter. Agustiarini et al. [147] estimate the coefficient of boiling heat transfer by fusing the ANN and experimental techniques. Experimental data is used to train and test the ANN. ANN estimates the boiling heat transfer coefficient to have a hidden layer and 16 input features. Evaluation of flow properties and heat transfer for a wavy channel with a turbulent flow is presented by Tahmasebian et al. [148], who compared the result generated with the modified SST (k-ω) turbulence model and RANS (Reynolds-averaged Navier-stokes) model and calculated the average Nusselt number with better accuracy. Hydrogen in the liquid state is stored at a temperature of 20 Kelvin; a slight heat leak to the storing tank may trigger the flow-boiling [149]. Huan Yang et al. [150] used a data-driven technique to predict the coefficient of heat transfer for flow boiling accurately. Batteries (Li-ion) are actively playing a significant role in powering the modern era of transportation and acting as an import and source of energy for other purposes. Charging and discharging generate heat due to the involved chemical processes. If these are not handled effectively, they can lead to a crucial harmful event (i.e., Thermal runaway), reduced lifespan, etc. Since data-driven analysis can handle large datasets, Miaari et al. [151] have suggested an ML method, predicted the battery's temperature, and effectively applied it in the thermal management system.

The development of flaws like cracks in continuous casting negatively impacts the efficiency of the steelmaking process and the quality of cast products. Phenomenological quality criteria calculated using a solidification and microstructure model called Inter-Dendritic Solidification (IDS) have previously been used to assess the risks and pinpoint the underlying reasons for defect formation. The method offered by Norrena et al. [152] used a basic understanding of fault generation in continuous casting and is computationally efficient. Combining IDS with a heat-transfer model simulates the process of continuous casting. The simulations use measured compositions, and the labels used for categorization are flaws reported at a steelmaking plant. To forecast transverse cracking in peritectic C–Mn, low-carbon B–Ti micro-alloyed, and peritectic Nb–micro-alloyed steels, logistic regression, decision trees, and Gaussian Naïve Bayes classifiers are created.

1.9.12 Predicting critical heat flux location

The site of critical heat flux (CHF) forecast is a complex problem in the power generation sector. Critical heat flux is affected by several variables, including the conditions of operation and geometric parameters. The dependency on these variables demonstrates a nonlinear relationship. To forecast the location of CHF, Kumar et al. [153] created various machine-learning models. He determined which model was most accurate based on comparisons with experimental data. Because experimental data spanned various operating settings, it was chosen for testing and training machine learning models. The models' performance is maximized through the application of hyperparameter adjustment. The artificial neural network (ANN) model outperforms all other developed models with both training and test datasets. A different model by Mudawar et al. [154] for the prediction of critical heat flux and heat transfer in micro-gravity and under the influence of earth's gravity for flow boiling using statistical analysis ANN is developed for the study of a number of input parameters that predicted Nusselt number.

1.9.13 Estimation of properties of nano fluids

Nanofluids (dispersion of nano-sized particles with base fluid) and colloidal nanoparticle solutions are receiving significant attention. Researchers are impressed with their enhanced characteristics (thermal, magnetic, or other properties) [155], being used in electronics cooling systems [156,157] and in biomedical fields [158,159]. Water-based Fe_3O_4–SiO_2 hybrid nanofluids were created by Alklaibi et al. [160], and the thermophysical parameters were calculated through experimentation. Using the experimental data, the Bayesian regularization-ANN analysis accurately predicts the friction factor and thermal performance factor. By applying the technique, the multi-linear is employed to create the friction factor based on the experimental data. Iron oxide and gold nanoparticles are essential in cancer therapy. Thus, the study of entropy production to quantify energy dissipation in biological systems is gaining attention from biomedical engineers and physicians. Using an MLP feed-forward backpropagation ANN and the Levenberg-Marquard algorithm, Jakeer et al. [161] created an intelligent numerical computing solver to interpret the Cattaneo-Christov heat flux model and show how entropy production and melting heat transfer affect the ferrohydrodynamic flow of the Fe3O4-Au/blood Powell-Eyring hybrid nanofluid. The heat transfer performance of a water-based nanofluid and carbon nanotube combination in a horizontal

tube with a fixed wall temperature and turbulent flow, investigated by Ullah et al. [162], is examined through experimentation. To propose a new correlation for the calculation of influential parameters on the heat transfer rate, such as the Nusselt number, friction factor, and overall thermal performance of heat exchangers, while using nanofluid and twisted tape simultaneously, experimental results will be utilized in an artificial neural network model.

Chapter-2

2. METHOD AND PROBLEM FORMULATION

There is transformative change in results across various domains, including cognitive science, genomics, and image recognition, due to the growth in computing power and some marvelous developments in machine learning and data-handling techniques. However, data collection costs need to be inflated while studying the biological, physical, and engineering systems. We often face the challenge of making decisions and drawing conclusions under partial or semi-partial information. Even the well-equipped state-of-the-art machine learning algorithms fail to guarantee convergence in this limited data domain.

Training neural network algorithms to identify the nonlinear relation or pattern between considerable dimensional input and output data is a little naïve. This generating relations between the pair mentioned above of data doesn't consider the pre-existing knowledge or principles of physical law that drive the system's dynamics. The principle of defined problem in the domain of science and engineering acts as a modifier that modifies the relation to an extent. Concatenating such information in existing neural networks amplifies desirable solutions even when a few training examples are available. Including the information to model the data-driven and physics-informed machine learning algorithms is presented in various recent works [163–165]. In these works, researchers have used the Gaussian regression process to get the outcome representation tailored to produce a linear operator, which infers the solution accurately [166].

2.1 Physics Informed Neural Network

The physics-informed neural network is one of the popular methods/techniques in the scientific community for solving the ODEs/PDEs. It is a class of neural networks that combines the physics behind the problem and the power of algorithms. Its flexibility and fantastic ability to approximate the function make it robust. Automatic differentiation, the algorithm used for the computation of derivatives, is

used to represent all the differential operators; hence, there is no need for a mesh generation, as in the case of computational fluid dynamics, i.e., CFD solver [167]. Physics-informed neural networks are not the replacement for computational fluid dynamics, but this technique performs more efficiently and more accurately than any computational fluid dynamics solver [168]. This shows that a physics-informed neural network can be used to analyze nonlinear heat transfer problems by combining knowledge of physics and neural networks and indicates that PINNs are effective tools for analyzing more complex problems. This method has been proposed in much recent research work, and it either belongs to the direct or forward method of heat transfer [169,170] or the inverse process of heat transfer [171]. Based on available data, problems can be classified as discrete-time and continuous-time models. In the forward type (as shown in Fig. 2.1) of heat transfer problem, the material's thermal conductivity (which may be different in different directions) is generally given, and the temperature distribution within the body is asked to be found. In general, forward kind of problems consume the knowledge of computational fluid dynamics or computational heat transfer, i.e., CHT, which converts the differential equation into a simple difference equation, or, in simple words, calculus problems are converted into algebraic problems. That makes the computation or result generation expansive. Problems associated with the kind of inverse method (Fig. 2.2) of heat transfer generally the distribution of temperature is given and asked to find the body's thermal conductivity (or any other properties).

Figure 2.1:Generalized forward model

Figure 2.2: Generalized inverse model

An advanced approach to solving computational models, including differential equations, is termed a physics-informed neural network (PINN). A generalized form is shown in Fig. 2.3. It relies on methods that include the artificial neural network and the physics or the governing equation of a specified problem. This work uses PINN training by optimization techniques on simple one-dimensional heat-transfer problems, namely slabs and fins with Dirichlet boundary conditions. This work provides a valuable approach for using PINN for different heat transfer problems. This work presents the main steps to developing such a neural network. For this, Python (version 3.9) and Keras library (version 3) based on TensorFlow (a scientific package) are used in the Jupyter Notebook environment.

2.2 Key concept of PINN

Integration of physics: PINNs incorporate the physical system's governing equations (e.g., PDEs) directly into the neural network training process. This is done by including the residuals of these equations in the loss function, ensuring that the network's predictions are consistent with the underlying physical laws. By embedding these laws into the training process, the neural network is constrained to learn physically consistent solutions, even in regions with sparse or no data. Conventional neural networks might need help to generalize well outside the range of training data. Still, PINNs ensure that predictions adhere to known scientific principles, leading to better generalization in unseen scenarios.

Loss function: The loss function in PINNs is augmented to include terms that enforce the satisfaction of the PDEs, boundary conditions, and initial conditions. This typically involves computing derivatives of the network's output with respect to its inputs using automatic differentiation. The standard loss function in neural networks is typically based on the difference between predictions and observed data. In PINNs, the loss function is augmented to include terms that enforce the satisfaction of the physical laws (e.g., PDEs). This often involves calculating the residuals of the PDEs using the network's predictions.

Forward and inverse problem

Forward problem: Given initial and boundary conditions, PINN can predict the solution to the PDE.

Inverse problem: PINNs can infer unknown parameters in the PDEs from observed data.

2.3 Problem formulation with PINN

1. Neural network architecture: A neural network is used to approximate the solution of the PDE. The network takes spatial and/or temporal coordinates as input and outputs the predicted value of the field variable (e.g., temperature, pressure).

2.Automatic differentiation: Automatic differentiation, or backpropagation, is used to compute the required derivatives of the network's output concerning its input. These derivatives are desirable for evaluating PDE residuals. Tools like TensorFlow or PyTorch offer automatic differentiation, which allows for the efficient calculation of derivatives of the neural network's output with respect to its input. These derivatives compute the PDE residuals, which are then incorporated into the loss function.

3. Loss function composition: The loss function typically includes:

a. Data loss: The discrepancy between the network predictions and any available data points.
b. PDE residual loss: Residual of PDE evaluated using the network's predictions
c. Boundary condition loss: The discrepancy between the network predictions and the specified boundary condition.

Consider a PDE of the form $\aleph(u) = f$, where, $\aleph$ is a differential operator, u is the unknown solution, and f is the source term. The augmented loss function in a PINN is

$$\boldsymbol{Loss} = \boldsymbol{Loss_{Data}} + \boldsymbol{Loss_{PDE}} + \boldsymbol{Loss_{BC}} + \boldsymbol{Loss_{IC}} \qquad (2.1)$$

Where

$Loss_{Data}$ is standard data loss i.e. mean squared error between prediction and truth value or observed data.

$Loss_{PDE}$ is measure of how well the network's prediction satisfies the PDE, i.e. mean square error of the PDE residual.

$Loss_{BC}$ enforces the boundary conditions, i.e. mean square error at the boundary points.

$Loss_{IC}$ enforces the initial condition, i.e. mean square error associated with initial condition.

4. Training: The network is trained using gradient-based optimization to minimize the combined loss function. During training, the network adjusts its parameters to satisfy the PDE and boundary conditions as closely as possible.

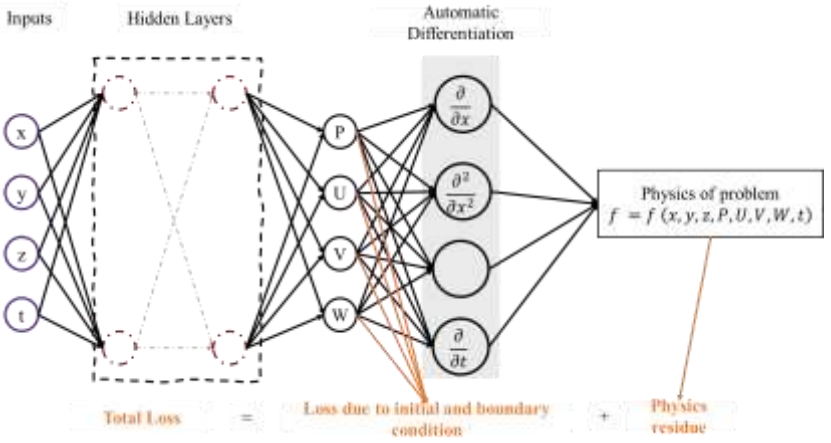

Figure 2.3: Generalized form of physics informed neural network.

2.4 Implementation and training of a physics-informed neural network

Training of problem, based on PINN to solve the PDE associated with heat transfer with specified boundary and initial condition was forced in Python (V3.9), using Tensor Flow, Keras, and other popular libraries such as numpy, matplotlib, scipy, etc. Training of such a neural network is based on selecting random batches for input variables (x, t) in each epoch and tries to minimize the loss function. This is obtained by the optimizer (built-in keras), which obtains weights and biases in the neural network, which satisfy heat transfer PDE.

Neural networks are known as universal approximators [172], i.e., they can tackle nonlinear problems without considering any prior assumption, linearization, or local time-stepping. We extract the power of automatic differentiation [173], one of the most valuable techniques in scientific computing, to differentiate a neural network concerning its input coordinates and model parameters to obtain physics-informed neural networks. Though simple, it is a robust development in physics, biology, and many other research domains.

Let's consider a parametrized partial differential equation of the general form as

$$\boldsymbol{u_t} + \aleph_{\boldsymbol{x}}\boldsymbol{u} = \boldsymbol{0}, \boldsymbol{x} \in \boldsymbol{\Omega}, \boldsymbol{t} \in [\boldsymbol{0}, \boldsymbol{t}] \tag{2.2}$$

Here

$u\,(t,x)$ denotes the latent or hidden solution

$\aleph[\cdot]$ is a non-linear differential operator, and Ω is subset of $\mathbb{R}^D$.

In continuous time models we define $f\,(t,x)$ to be given by:

$$\boldsymbol{f} := \boldsymbol{u_t} + \aleph_{\boldsymbol{x}}[\boldsymbol{u}] \tag{2.3}$$

and processed by approximating $u\,(t,x)$ by a deep neural network. This assumption results in a Physics Informed Neural Network $f\,(t,x)$.

The work aims to set the foundation for the new computation and modeling paradigm that explores deep learning. Let's consider our first study:

2.5 Study I: Heat transfer through a slab

Transient heat conduction is essential in many engineering scenarios, like heating and cooling metal billets, quenching metal for heat treatment, cooling internal combustion engines, and operating heat exchange units in power plants. Consider a one-dimensional heat conduction through a slab as shown in Fig. 2.4. This slab is maintained at a constant temperature at both ends as a boundary condition, and a sinusoidal initial condition is also imposed with the problem, as shown in the figure below. The subsequent part consists of two solutions: (i) analytical method, which is followed by most of the researchers [174,175], and (ii) machine learning-based physics-informed neural network method. After applying the physics in the neural network, i.e., PINN, only 16 training and 100 test points are used. The architecture used for this neural network consists of [2, 50*3, 1], where the first element in the list, i.e., 2, indicates the number of neurons in the input layer, the second element in the list, i.e., 50*3 indicates the hidden layer which means there are three hidden layers with each containing 50 neurons and the last element indicates the number of neurons that is producing the output. This neural network takes space dimension and time, i.e., x and t, as input parameters and produces temperature, i.e., T as an output.

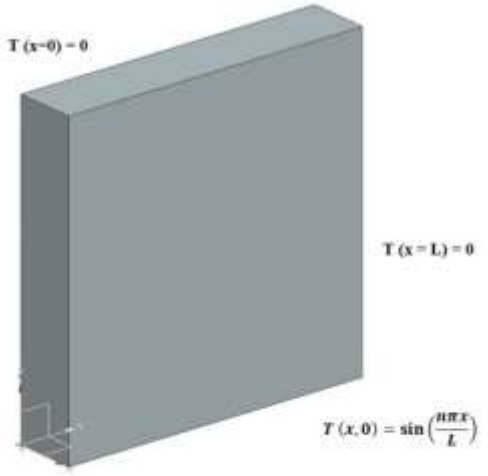

Figure 2.4: Slab with specified boundary and initial condition.

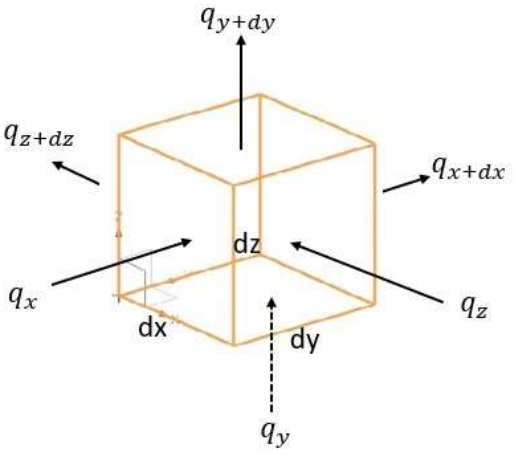

Figure 2.5: A differential control volume.

Consider a differential control volume element (Fig. 2.5) whose dimensions are dx, dy, and dz in X, Y, and Z direction, respectively, for a stationary, homogeneous isotropic solid with heat generation within the body, as shown in the figure.

Differential control volume $dv = dx\ dy\ dz$

Differential control mass $dm = \rho\ dv$

From 1'st law of thermodynamics i.e. conservation of energy:

$$\begin{Bmatrix} Net\ rate\ of\ heat \\ transfer\ to\ the\ body \end{Bmatrix} + \begin{Bmatrix} Rate\ of\ heat \\ generation \end{Bmatrix} = \begin{Bmatrix} Rate\ of\ increase\ of \\ internal\ energy\ of\ body \end{Bmatrix} \tag{2.4}$$

$$\delta\dot{Q} + \delta\dot{q} = \frac{d(U\ dm)}{dx} \tag{2.5}$$

Net rate of heat transfer in terms of heat rate in and out of control volume

$$\delta\dot{Q} = (q_x - q_{x+dx}) + \left(q_y - q_{y+dy}\right) + (q_z - q_{z+dz}) \tag{2.6}$$

Entering heat rates are defined as follow:

$$q_x = -K\ A_x \frac{\partial T}{\partial x} \tag{2.7}$$

Where $A_x = dy\ dz$

Exiting heat rates are defined as follows:

$$q_{x+dx} = q_x + \frac{\partial q_x}{\partial x} dx \tag{2.8}$$

Following the same analogy for the y and z direction, we get

$$\frac{\partial}{\partial x}\left(K\frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial x}\left(K\frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial x}\left(K\frac{\partial T}{\partial x}\right) + \dot{q} = \rho C\frac{\partial T}{\partial t} \tag{2.9}$$

In vectorial form

$$\nabla \cdot (K\nabla T) + \dot{q} = \rho\ C\ \frac{\partial T}{\partial t} \tag{2.10}$$

Assumption: Thermal conductivity is constant

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial x^2} + \frac{\dot{q}}{K} = \frac{1}{\alpha}\frac{\partial T}{\partial t} \tag{2.11}$$

Where $\alpha = \frac{K}{\rho \mathrm{C}}$

Assumption: No heat generation

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial x^2} = \frac{1}{\alpha}\frac{\partial T}{\partial t} \tag{2.12}$$

Assumption: Heat transfer takes place in x-direction only

$$\frac{\partial^2 T}{\partial x^2} = \frac{1}{\alpha}\frac{\partial T}{\partial t}$$

Associated boundary and initial conditions are:

Boundary conditions are:

$$T\ (x = 0, t) = 0 \tag{2.14}$$
$$T\ (x = L, t) = 0 \tag{2.15}$$

Initial condition:

$$T\ (x, 0) = \sin\left(\frac{n\pi x}{L}\right) \tag{2.16}$$

Changing the governing equation, boundary conditions and initial condition into non-dimensional form, leads to:

$$\frac{\partial^2 T^*}{\partial x^{*2}} = \frac{\partial T^*}{\partial t^*} \tag{2.17}$$

Where $x^* = \frac{x}{L}$; $T^* = \frac{T - T_\infty}{T_o - T_\infty}$; and $t^* = \frac{\alpha\, t}{L^2}$

Non-dimensional form of boundary condition becomes:

$$T^*(x^* = 0, t^*) = 0 \tag{2.18}$$
$$T^*(x^* = 1, t^*) = 0 \tag{2.19}$$

Non-dimensional form of initial condition becomes:

$$T^*(x^*, t^* = 0) = sin(n\pi x^*) \tag{2.20}$$

Applying method of separation of variables, we get

$$T^*(x^*, y^*) = e^{-\lambda^2 t^*} \sin(n\pi x^*) \tag{2.21}$$

Changing the temperature back into dimensional form we get,

Since,

$$T = T_\infty + (T_0 - T_\infty)T^* \tag{2.22}$$

$$T = T_\infty + (T_0 - T_\infty)\left[e^{-\lambda^2 t^*} sin(n\pi x^*)\right] \tag{2.23}$$

2.6 Applying physics-informed neural network to our problem

For the development of a neural network, in association with our problem of the slab, space dimension x and time t are considered the input for the neurons in the input layer depicted in Fig. 2.6. The governing equation, boundary conditions and initial condition for the above-mentioned problem

$$\frac{\partial^2 T^*}{\partial x^{*2}} = \frac{\partial T^*}{\partial t^*} \tag{2.24}$$

$$T^*(x^* = 0, t^*) = 0 \tag{2.25}$$

$$T^*(x^* = 1, t^*) = 0 \tag{2.26}$$

$$T^*(x^*, t^* = 0) = sin(n\pi x^*) \tag{2.27}$$

Transformation for the PINN:

$$u_t + \aleph_x u = 0, \qquad x \in [0,1], \qquad t \in [0,\ t] \tag{2.28}$$

$$u(\ t = 0, x) = sin(n\pi x) \tag{2.29}$$

$$u(x = 0, t) = 0 \tag{2.30}$$

$$u(x = 1, t) = 0 \tag{2.31}$$

Let us define $f\ (t,x)$ to be given by

$$f\ :=\ u_t\ +\ \aleph_x u \tag{2.32}$$

and proceed by approximating $u(\ t,x)$ by a deep neural network. The shared parameter between the neural network $u(\ t,x)$ and $f\ (t,x)$ can be learned by minimizing the mean squared error loss

$$MSE\ =\ MSE_u\ +\ MSE_f \tag{2.33}$$

Where,

$$MSE_u\ =\ \frac{1}{N_u}\sum_{i\ =\ 1}^{N_u}\left|u\left(t_u^i,x_u^i\right)\ -\ u^i\right|^2 \tag{2.34}$$

And

$$MSE_f\ =\ \frac{1}{N_f}\sum_{i\ =\ 1}^{N_f}\left|f\left(t_f^i,x_f^i\right)\ \right|^2 \tag{2.35}$$

Here $\left\{t_u^i,x_u^i,u^i\right\}_{i\ =\ 1}^{N_u}$ denotes the initial and boundary training data on $u(\ t,x)$ and $\left\{t_f^i,x_f^i\right\}_{i\ =\ 1}^{N_u}$ specify the collection points for $f\ (t,x)$. The loss MSE_u corresponds to the initial and boundary data while MSE_f enforce the structure imposed by the above-mentioned Heat equation at a finite set of collection points.

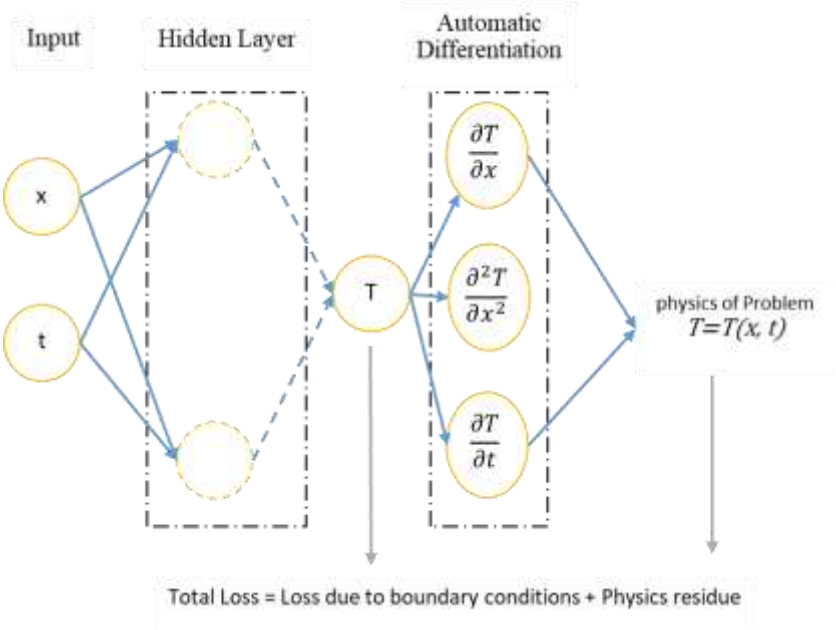

Figure 2.6: Intuition of PINN used in study I.

2.7 Study II: Thermal analysis of fin

Extended surfaces, also known as fins, are widely used in heat transfer, increasing the heat transfer rate from base to surroundings. These extended surfaces are mostly used when the convective heat transfer rate between base and environment fluids is low. Most simple geometry (constant cross-section area) with an axial coordinate system is considered for our analysis. Mathematical simplicity of one-dimensional, with steady-state heat transfer, exists in several engineering or research domains. Figure 2.7 represents a fin with a constant cross-section, which is maintained at a constant temperature at both ends. All the other parameters associated with design, environment, and material are also expressed in the figure. Applying the physics in the neural network, i.e., PINN, only 18 training and initially 100 test points were used. The architecture used for this neural network consists of [1, 50*3, 1], where the first element in the list, i.e., 1, indicates the number of neurons in the input layer, the second element in the list, i.e., 50*3, indicates the hidden layer which means there are three hidden layers with each containing 50 neurons and the last element indicates the number of neurons that is producing the output. This neural network takes space dimension, i.e., x as the input, and it produces temperature as output.

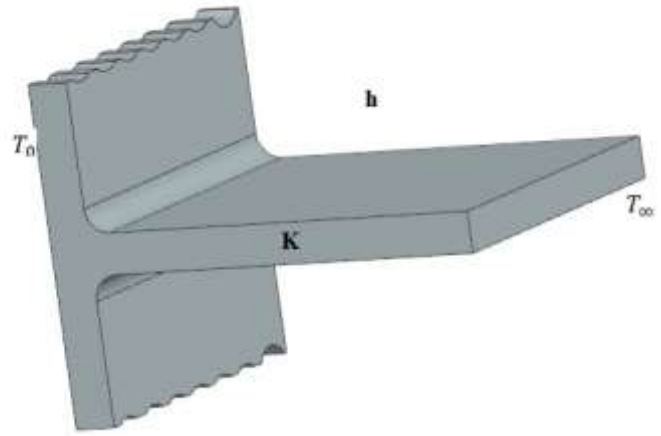

Figure 2.7: Extended surface with specified boundary condition

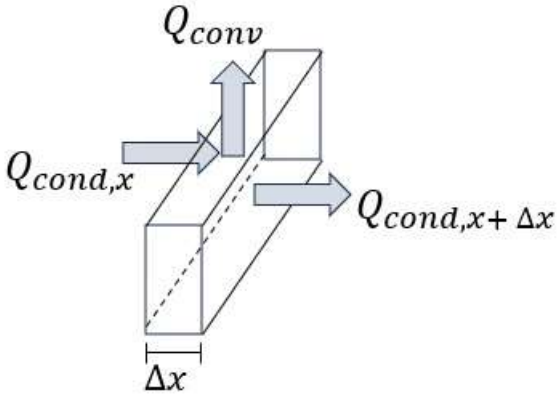

Figure 2.8: A differential control volume.

Deriving the governing equation for fin:

Consider a differential element of fin, at a distance x, of length Δx, cross-sectional area of A and perimeter of P, shown in above Fig. 2.8.

$$\left\{\begin{matrix}Rate\ of\ heat\\ conduction\\ into\ the\ element\ at\ x\end{matrix}\right\} = \left\{\begin{matrix}Rate\ of\ heat\\ conduction\\ from\ the\ element\\ at\ x+\ \Delta x\end{matrix}\right\} + \left\{\begin{matrix}Rate\ of\ heat\ convection\\ from\ the\ element\end{matrix}\right\} \quad (2.36)$$

$$\dot{Q}_{cond,x} = \dot{Q}_{cond,x+\Delta x} + \dot{Q}_{conv}$$
$$\text{where, } \dot{Q}_{conv} = h(P\Delta x)(T - T_\infty) \quad (2.37)$$

Substituting and dividing by Δx, we get

$$\frac{\dot{Q}_{cond,x+\Delta x} - \dot{Q}_{cond,x}}{\Delta x} + hP(T - T_\infty) = 0 \quad (2.38)$$

As $\Delta x \rightarrow 0$

$$\frac{d\dot{Q}_{cond}}{dx} + hP(T - T_\infty) = 0 \tag{2.39}$$

From Fourier's law of heat conduction $\dot{Q}_{cond} = -KA\frac{dT}{dx}$, we get

$$\frac{d}{dx}\left(KA\frac{dT}{dx}\right) - hP(T - T_\infty) = 0 \tag{2.40}$$

The governing equation of fin:

$$\frac{d^2T}{dx^2} - \frac{hP}{KA}(T - T_\infty) = 0 \tag{2.41}$$

Specified boundary conditions are:

$$T\ (x = 0) = 350\ °C \text{ and} \tag{2.42}$$

$$T\ (x = L) = 25\ °C \tag{2.43}$$

In dimensionless form, the governing equation becomes:

$$\frac{d^2T^*}{dx^{*2}} - \frac{hP}{KA}L^2(T^* - T_\infty) = 0 \tag{2.44}$$

Boundary condition in dimensionless form

$$T^*\ (x^* = 0) = 1 \text{ and} \tag{2.45}$$

$$T^*\ (x^* = 1) = 0 \tag{2.46}$$

Solving the governing equation for the specified boundary condition gives

$$T^* = C_1 e^{mL\,x^*} + C_2 e^{-mL\,x^*} \tag{2.47}$$

Where, $C_1 = \frac{e^{-mL}}{e^{mL} - e^{-mL}}$ and $C_2 = \frac{e^{mL}}{e^{mL} - e^{-mL}}$

2.8 Applying physics informed neural network to our study

For the development of a neural network, in association with our problem of fin, space dimension x is considered the input for the neurons present in the input layer depicted in Fig. 2.9. The governing equation and boundary conditions for the above-mentioned problem

$$\frac{d^2T^*}{dx^{*2}} - \frac{hP}{KA} L^2(T^* - T_\infty) = 0 \tag{2.48}$$

$$T^*(x^* = 0) = 1 \tag{2.49}$$

$$T^*(x^* = 1) = 0 \tag{2.50}$$

Transformation for the PINN:

$$u_t + \aleph_x u = 0, \qquad x \in [0,1] \tag{2.51}$$

$$u(x = 0, t) = 0 \tag{2.52}$$

$$u(x = 1, t) = 0 \tag{2.53}$$

Let us define $f\ (t, x)$ to be given by

$$f := u_t + \aleph_x u \tag{2.54}$$

and proceed by approximating $u(x)$ by a deep neural network. The shared parameter between the neural network $u(x)$ and $f\ (x)$ can be learned by minimizing the mean squared error loss

$$MSE = MSE_u + MSE_f \tag{2.55}$$

Where,

$$MSE_u = \frac{1}{N_u}\sum_{i=1}^{N_u}\left|u(x_u^i) - u^i\right|^2 \text{ and} \tag{2.56}$$

$$MSE_f = \frac{1}{N_f}\sum_{i=1}^{N_f}\left|f(x_f^i)\right|^2 \tag{2.57}$$

Here $\{x_u^i, u^i\}_{i=1}^{N_u}$ denotes the boundary training data on $u(x)$ and $\{x_f^i\}_{i=1}^{N_u}$ specifies the collection points for $f(x)$. The loss MSE_u corresponds to the boundary data while MSE_f enforces the structure imposed by the above-mentioned fin equation at a finite set of collection points.

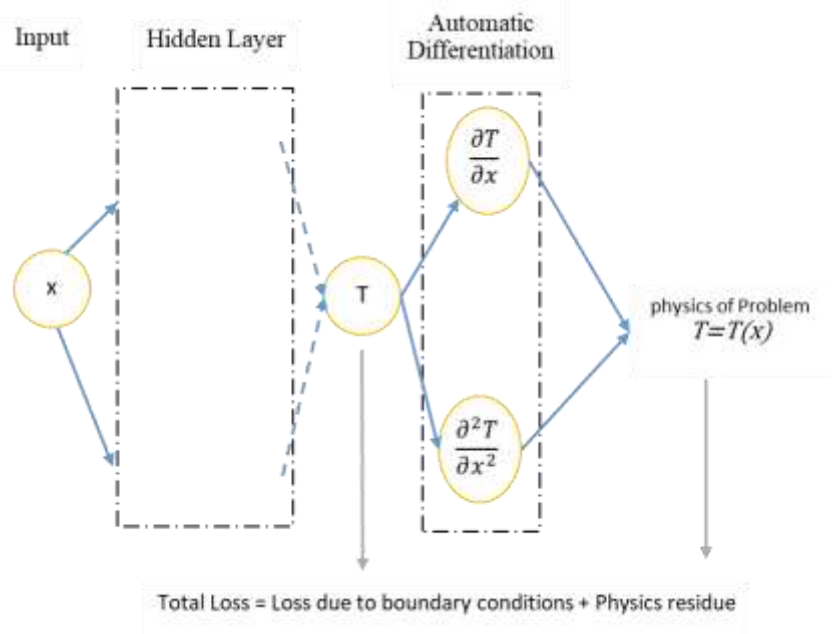

Figure 2.9: Intuition of PINN used in study II.

Chapter-3

3. RESULTS AND DISCUSSION

Let's interpret the plot obtained from the model suggested in chapter 2 of this work. Figures 3.1, 3.2, and 3.3 display the training loss and test loss over a number of steps during a machine learning training process. Below are explanations of what each component indicates.

The line representing the training loss indicates an error on the training dataset. It shows how well the model is learning to fit the training data. It shows a decreasing trend as the model gets trained, i.e., the model is being trained well. Test loss on the figures mentioned above indicates an error on the test dataset. Its decreasing trend shows how well the model generalizes the unseen data. Its closeness with the training line shows the consistency and relatively small gap between train and test loss. To conclude, plots indicate a well-performing model with the following characteristics:

a. Consistently decreasing training and testing losses.
b. A small gap between training and testing loss suggests good generalization.
c. An improving test metric.

Figure 3.2 shows that train and test losses are decreasing over time, which suggests that the model is learning the training data very well. However, if the test metric is also reduced, the model's performance worsens on unseen data, possibly due to overfitting. This decreasing trend in the test metric is a cause for concern because it suggests that the model's performance might not generalize well to new data.

Figure 3.3 represents the decreasing trend of test and train loss with time. It means our model is being trained and acting well. For a model with nearly constant or stable value, demonstrate a model that maintains consistent performance on unseen data despite some training and test loss fluctuations. This stability suggests that the model is well-regularized and capable of effectively generalizing new data, a vital indicator of a robust machine-learning model.

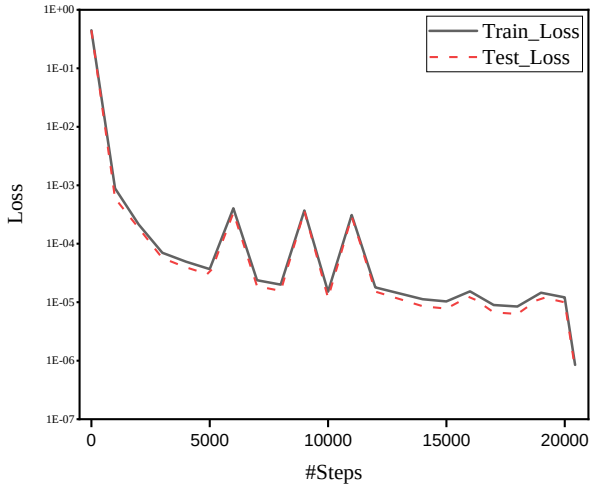

Figure 3.1: Train and test loss vs steps (for slab).

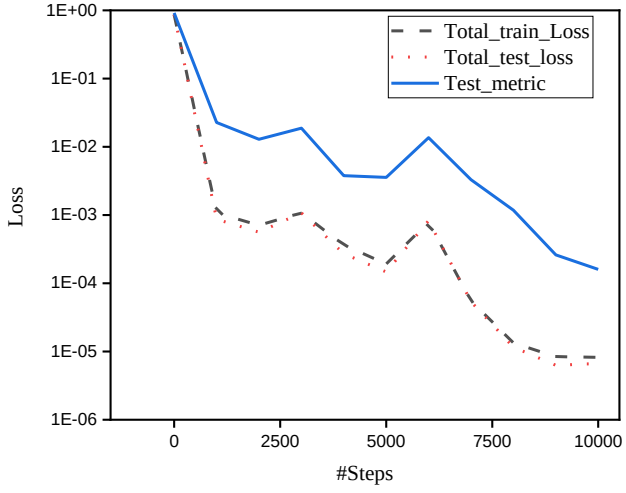

Figure 3.2: A bad model: Train and test loss vs steps (for fin).

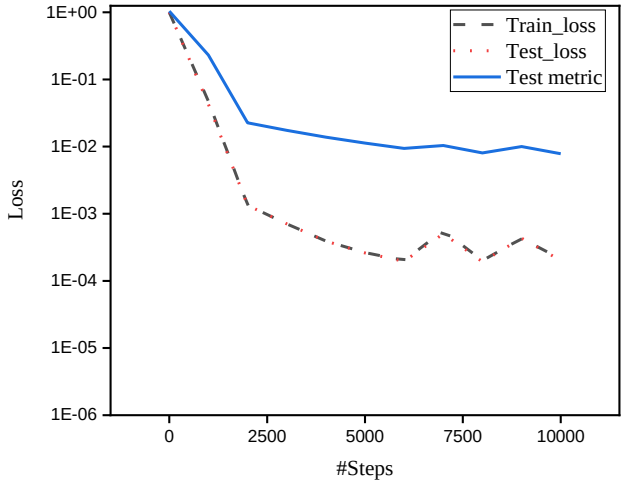

Figure 3.3: A good performing model, Train and test loss vs steps (for fin).

3.1 Results associated with study of a slab

A physics-informed neural network-based solution presented for one-dimensional heat conduction through a slab with constant boundary (i.e., Dirichlet boundary) condition with the sinusoidal initial condition is presented in this work. The analytical solution [176], based on the method of separation of variables, of the problem mentioned above is also compiled and used for validation purposes. Glorot uniform is used as initializers (assigns initial weights before the training begins). For spatial dimension sine function and for time dimension exponential function is used as activation functions in our neural network. In a based model, Keras built-in Adam optimizer with a learning rate of 0.001 and L2 relative error is assigned as metric to minimize the loss function. This model is trained for more than 20,000 iterations. Initially, the model is compiled and trained using Adam optimizer, a stochastic gradient descent method that works well with large-scale datasets. After initial training with Adam, again, the model is compiled using L-BFGS, i.e., Limited-memory Broyden-Fletcher-Goldfarb-Shanno, a kind of deterministic optimization algorithm that provides an accurate and precise solution in the case of PDEs. Using Adam

and L-BFGS helps quickly find a region near the optimal solution, while L_BFGS converges to the minimum loss function more accurately. A decrement in training data loss (5.45e-06 for the best model) indicates the learning ability of the model, i.e., the model is learning to the approximate solution. The test loss is evaluated on the test data; it is a measure of how well the model generalizes to unseen data. L2 relative error gives the normalized measure of error among predicted and relative solutions. Our model's mean residual and L2 relative error are 0.00058 and 0.00153, respectively. Data presented in Table 3.1 compares the predicted temperature with the exact temperature that our model generates. Data presented in this table states the temperature in the slab at different positions along its length (i.e., in the x-direction) with varying instances of time. Using this model, a Table 3.1 is created that shows the distribution of temperatures (exact and predicted) concerning position and time. Using this table, a 3D plot, as shown in Fig. 3.4, is plotted, showing the temperature distribution concerning space and time. Table 3.2 shows the exact temperature and predicted temperature at different instances of time at position x = 0.4 from the slab's left end. Table 3.3 represents the exact temperature and predicted temperature to varying positions at step t = 0.5.

Data in Table 3.2 is used to plot Fig. 3.5, which shows the decreasing trend of temperature with an increase in time. This trend is exponential. During the initial period, the plot shows a higher gradient, i.e., approximately 70 % of temperature reduces in the first 30 % of the time. As time progresses, the gradient starts decreasing. Figure 3.6 is plotted using data available in Table 3.3. This plot shows the temperature distribution with respect to space, i.e., along x direction. The distribution is sinusoidal. It indicates the positive slope for approx. The first half of the slab and the slope change to negative for the latter half of the slab are obtained. The sinusoidal behavior of temperature distribution is maintained even as time moves forward, which is shown in Fig. 3.4. As time progresses, the maximum temperature approaches a particular point and starts reducing.

Table 3.1: Variation in exact and predicted temperature with space and time in slab.

X	Time	Exact Temperature	Predicted Temperature
0.0000000000	0.0000000000	0.0000000000	-0.0001271814
0.2000000000	0.0000000000	0.5877852523	0.5878090262
0.4000000000	0.0000000000	0.9510565163	0.9511439204

0.6000000000	0.0000000000	0.9510565163	0.9510025382
0.8000000000	0.0000000000	0.5877852523	0.5877820849
1.0000000000	0.0000000000	0.0000000000	0.0000726730
0.0000000000	0.2000000000	0.0000000000	0.0000565499
0.2000000000	0.2000000000	0.2668784502	0.2669838071
0.4000000000	0.2000000000	0.4318184032	0.4318987727
0.6000000000	0.2000000000	0.4318184032	0.4319103956
0.8000000000	0.2000000000	0.2668784502	0.2669804096
1.0000000000	0.2000000000	0.0000000000	-0.0000050962
0.0000000000	0.4000000000	0.0000000000	-0.0000082254
0.2000000000	0.4000000000	0.1211736887	0.1211841553
0.4000000000	0.4000000000	0.1960631468	0.1960862726
0.6000000000	0.4000000000	0.1960631468	0.1960507482
0.8000000000	0.4000000000	0.1211736887	0.1211010069
1.0000000000	0.4000000000	0.0000000000	-0.0001224577
0.0000000000	0.6000000000	0.0000000000	-0.0001854002
0.2000000000	0.6000000000	0.0550177911	0.0549526960
0.4000000000	0.6000000000	0.0890206560	0.0889871418
0.6000000000	0.6000000000	0.0890206560	0.0890056789
0.8000000000	0.6000000000	0.0550177911	0.0550205261
1.0000000000	0.6000000000	0.0000000000	0.0000563413
0.0000000000	0.8000000000	0.0000000000	-0.0001919717
0.2000000000	0.8000000000	0.0249803185	0.0248077810
0.4000000000	0.8000000000	0.0404190044	0.0403219461
0.6000000000	0.8000000000	0.0404190044	0.0403976142
0.8000000000	0.8000000000	0.0249803185	0.0250364840
1.0000000000	0.8000000000	0.0000000000	0.0000682324
0.0000000000	1.0000000000	0.0000000000	0.0006562322
0.2000000000	1.0000000000	0.0113420823	0.0115367025
0.4000000000	1.0000000000	0.0183518746	0.0183852166
0.6000000000	1.0000000000	0.0183518746	0.0182902515
0.8000000000	1.0000000000	0.0113420823	0.0112543404
1.0000000000	1.0000000000	0.0000000000	-0.0001673251

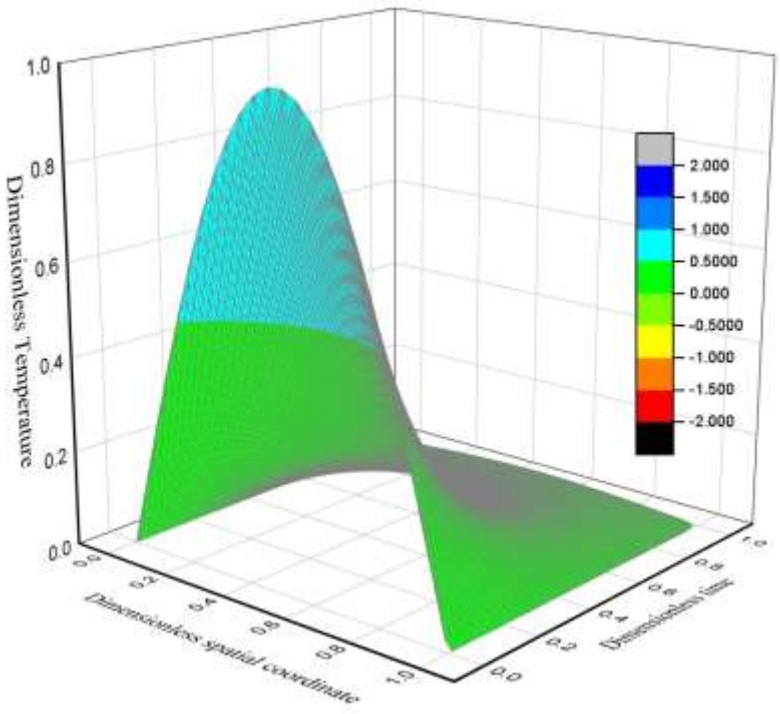

Figure 3.4: Distribution of temperature with space and time.

Table 3.2: Temperature versus Time (at x = 0.4) in slab.

Time	Exact Value	Predicted Value
0	0.951056516	0.95114392
0.03	0.844832376	0.844834983
0.06	0.750472481	0.750431836
0.09	0.666651707	0.666628182
0.12	0.59219293	0.592211843

0.15	0.526050505	0.526106298
0.18	0.467295571	0.467371404
0.21	0.415103015	0.415183067
0.24	0.368739881	0.368811905
0.27	0.327555077	0.327615321
0.3	0.29097023	0.291018069
0.33	0.25847157	0.258509219
0.36	0.229602707	0.229633018
0.39	0.203958227	0.203983262
0.42	0.181177996	0.181198224
0.45	0.160942104	0.160957173
0.48	0.142966372	0.142975107
0.51	0.126998362	0.126999661
0.54	0.11281383	0.112804756
0.57	0.100213578	0.10019277
0.6	0.089020656	0.088987142
0.63	0.079077879	0.079030216
0.66	0.070245618	0.070184007
0.69	0.062399838	0.062325969
0.72	0.055430358	0.05534552
0.75	0.049239304	0.049146578
0.78	0.043739733	0.043642685
0.81	0.038854413	0.038758367
0.84	0.034514738	0.034423247
0.87	0.030659763	0.03057979
0.9	0.027235354	0.027171746
0.93	0.024193419	0.024152651
0.96	0.02149124	0.021478429
0.99	0.019090869	0.019111887
1	0.018351875	0.018385217

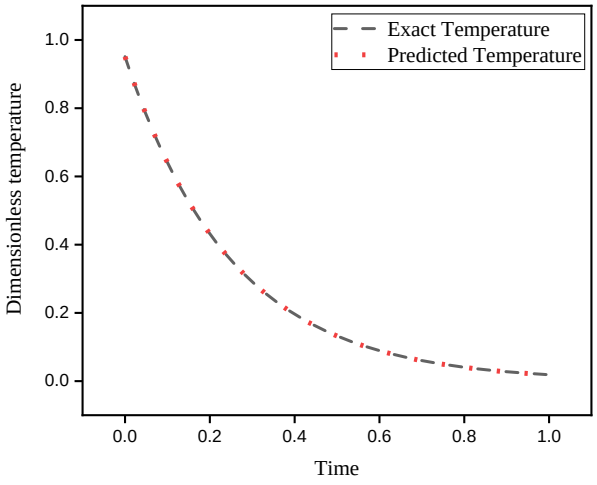

Figure 3.5: Temperature vs Time (at x=0.4).

Table 3.3: Temperature vs Position (at t = 0.5).

x	Exact Temperature	Predicted Temperature
0.0000000000	0.0000000000	-0.0000490397
0.0235294118	0.0102589383	0.0102211386
0.0470588235	0.0204618458	0.0204350948
0.0705882353	0.0305529978	0.0305361748
0.0941176471	0.0404772799	0.0404683053
0.1176470588	0.0501804892	0.0501791090
0.1411764706	0.0596096300	0.0596141219
0.1647058824	0.0687132037	0.0687216222
0.1882352941	0.0774414897	0.0774532408
0.2117647059	0.0857468172	0.0857598782
0.2352941176	0.0935838253	0.0935977995
0.2588235294	0.1009097109	0.1009242386
0.2823529412	0.1076844627	0.1076980680
0.3058823529	0.1138710794	0.1138834208
0.3294117647	0.1194357716	0.1194467098
0.3529411765	0.1243481471	0.1243570298
0.3764705882	0.1285813762	0.1285881549
0.4000000000	0.1321123384	0.1321160048
0.4235294118	0.1349217487	0.1349221021
0.4470588235	0.1369942633	0.1369912475
0.4705882353	0.1383185627	0.1383117884
0.4941176471	0.1388874141	0.1388766617
0.5176470588	0.1386977106	0.1386829466
0.5411764706	0.1377504883	0.1377319545
0.5647058824	0.1360509206	0.1360284537
0.5882352941	0.1336082900	0.1335821599
0.6117647059	0.1304359372	0.1304060072
0.6352941176	0.1265511885	0.1265183240
0.6588235294	0.1219752612	0.1219393462
0.6823529412	0.1167331472	0.1166944355
0.7058823529	0.1108534773	0.1108111590

0.7294117647	0.1043683641	0.1043227166
0.7529411765	0.0973132272	0.0972651541
0.7764705882	0.0897265991	0.0896750391
0.8000000000	0.0816499154	0.0815946907
0.8235294118	0.0731272883	0.0730694979
0.8470588235	0.0642052652	0.0641461164
0.8705882353	0.0549325752	0.0548726171
0.8941176471	0.0453598626	0.0453017205
0.9176470588	0.0355394101	0.0354847908
0.9411764706	0.0255248537	0.0254758894
0.9647058824	0.0153708894	0.0153306127
0.9882352941	0.0051329747	0.0051035881
1.0000000000	0.0000000000	-0.0000234395

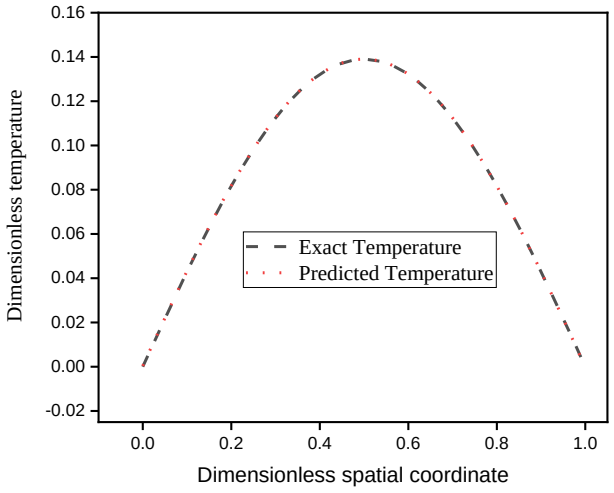

Figure 3.6: Temperature vs Space (at t=0.5).

3.2 Results associated with study of fin

The temperature distribution in the fin is plotted in Figs. 3.7, 3.8, and 3.9 for the mL value of 4, 1, and 0.5, respectively. These plots show the exactness of the predicted temperature. From the plot (figure 3.7), the temperature gradient decreases and increases in x. This is due to the reduction in conduction heat transfer and the increase in x-direction due to continuous convection loss through the fin surface. The main focus of fin design is to minimize the fin material, which impacts the manufacturing cost without compromising cooling effectiveness. Table 3.4 represents temperature distribution along the x-direction for the different values of mL = 4, mL = 1, and mL = 0.5. The maximum temperature value exists at the base, which is obvious and keeps decreasing exponentially as x increases. Figure 3.10 shows the comparison between the different values of mL. Interpretation of this figure leads to the following conclusions:

1. For any value of x, in the case of a small value of mL, the slope of the temperature gradient is always negative and nearly constant. That means loss in heat flux due to conduction heat transfer is always positive from left to right, i.e., increasing direction of x.
2. For the higher value of mL, the slope decreases from a higher negative value to a lower negative value.

Glorot uniform and tanh are used as initializers (assigns initial weights before the training begins) and activation functions for the neural network. In a based model, Keras built-in Adam optimizer with a learning rate of 0.001 and L2 relative error is assigned as metric to minimize the loss function. This model is trained for more than 10,000 iterations. A decrement in training data loss indicates the learning ability of the model, i.e., the model is learning to the approximate solution. The test loss is evaluated on the test data; it measures how well the model generalizes to unseen data. L2 relative error gives the normalized error measure among predicted and relative solutions. Our model's mean residual and L2 relative error are 0.00058 and 0.00153, respectively.

Table 3.4: Predicted and exact temperature distribution at different position of x for different value of mL.

	mL = 4		mL = 1		mL = 0.5	
X	**Exact Temperature**	**Predicted Temperature**	**Exact Temperature**	**Predicted Temperature**	**Exact Temperature**	**Predicted Temperature**
0	0.99999994	0.999928176	1	1.00012362	0.99999994	1.000052452
0.025025	0.949313402	0.949247897	0.967450857	0.967579782	0.973001003	0.973056018
0.05005	0.901005328	0.900948465	0.935507894	0.935642779	0.946154416	0.946211755
0.075075	0.854954898	0.854907811	0.904150546	0.904293478	0.919456005	0.919516087
0.1001	0.811046362	0.811008692	0.873359561	0.873512626	0.89290148	0.892964661
0.125125	0.769170105	0.769139171	0.843115628	0.843280613	0.866486847	0.866553545
0.15015	0.729220867	0.729194105	0.813399673	0.813577473	0.840207636	0.840278029
0.175175	0.691098928	0.691073835	0.784193039	0.784383357	0.814060211	0.814134002
0.2002	0.654708385	0.654683888	0.755477548	0.755679488	0.788040221	0.788117468
0.225225	0.619958341	0.619934678	0.727235317	0.727447033	0.762143612	0.762224078
0.25025	0.586761534	0.586739957	0.699448526	0.69966799	0.736366212	0.736449778
0.275275	0.555034876	0.555017054	0.67209971	0.672324598	0.710704207	0.710790336
0.3003	0.524698973	0.524686933	0.645171762	0.645400107	0.685153484	0.685241878
0.325325	0.49567759	0.495672941	0.618648052	0.618877828	0.65971005	0.659800649
0.35035	0.46789825	0.467902094	0.592511773	0.592741728	0.634369671	0.634462416
0.375375	0.441291213	0.44130373	0.566746533	0.566975892	0.609128833	0.609223485
0.4004	0.415789843	0.415810525	0.541336238	0.541564822	0.583983302	0.58408004
0.425425	0.391330242	0.391358107	0.516264856	0.516492963	0.558929324	0.559028029
0.45045	0.367851168	0.367884755	0.491517007	0.491745621	0.533962607	0.534063935
0.475475	0.34529373	0.345331222	0.467076808	0.467307419	0.509079814	0.509183764
0.500501	0.323601395	0.32364133	0.442929268	0.443163276	0.484276354	0.484383613
0.525526	0.302719921	0.302761078	0.419059098	0.419298619	0.45954901	0.459659666
0.550551	0.282596827	0.282638609	0.395451218	0.39569819	0.434893429	0.435008138
0.575576	0.263181865	0.263223857	0.372091204	0.372347653	0.410306036	0.410424978
0.600601	0.244426295	0.244468883	0.348964185	0.349231869	0.385782778	0.385906219
0.625626	0.226283163	0.226327553	0.326055676	0.326335937	0.361320138	0.361448318

0.650651	0.208706945	0.208754614	0.303351313	0.303645223	0.336913884	0.337046802
0.675676	0.191653699	0.191706493	0.280837029	0.281145275	0.31256026	0.312698096
0.700701	0.175080597	0.175140649	0.258498609	0.25882107	0.28825593	0.288398147
0.725726	0.158946186	0.159015477	0.236322105	0.236658409	0.263996661	0.264142931
0.750751	0.143210024	0.143290207	0.21429354	0.214642599	0.239778638	0.239928544
0.800801	0.112775587	0.112880617	0.170625359	0.170994833	0.191451311	0.191606998
0.825826	0.098001085	0.098118424	0.148958415	0.149335012	0.167334437	0.167491913
0.850851	0.083472133	0.083601013	0.127384752	0.127765894	0.14324373	0.143402249
0.875876	0.069152325	0.069291025	0.10589084	0.106274351	0.119175673	0.119334616
0.900901	0.055005804	0.055151928	0.084463298	0.084846884	0.095126212	0.095284916
0.925926	0.040997081	0.041148525	0.063088626	0.063470781	0.071091533	0.071249709
0.950951	0.027091056	0.027245577	0.041753441	0.042133164	0.047068	0.047225624
0.975976	0.013252929	0.013409648	0.020444393	0.020822208	0.023052037	0.023209166
1	1.49E-08	0.000159834	-5.96046E-08	0.000377238	0	0.000157449

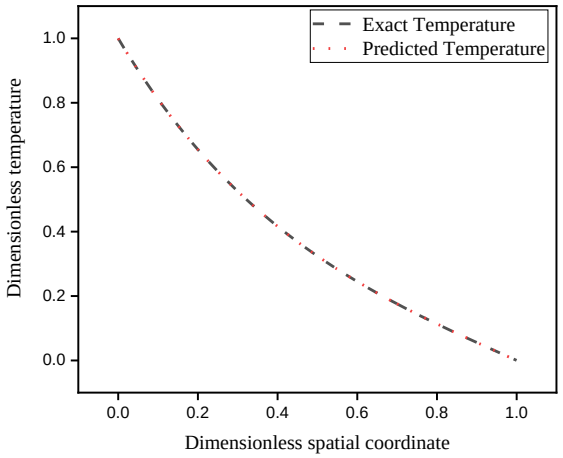

Figure 3.7: Dimensionless temperature vs Dimensionless space for mL = 4.

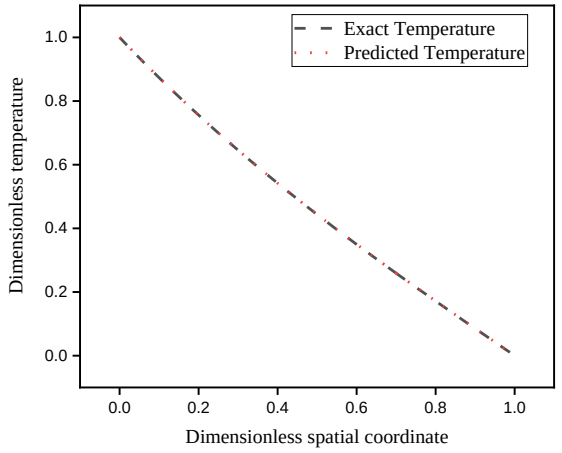

Figure 3.8: Dimensionless temperature vs dimensionless space for mL=1.

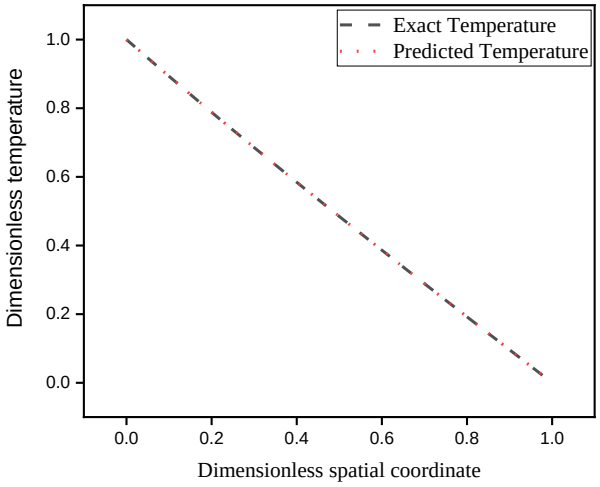

Figure 3.9: Dimensionless temperature vs dimensionless space for mL=0.5.

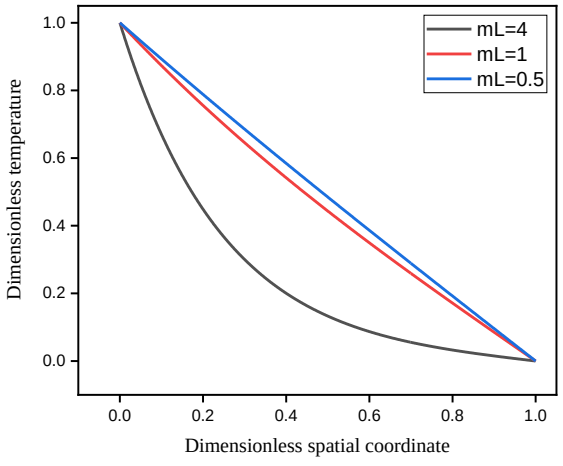

Figure 3.10: Non-dimensional temperature profile for different values of mL

Chapter-4

4. FUTURE SCOPE AND CONCLUSION

As a tool, machine learning can relate the dependent and independent variables using different algorithms. One of the most commonly used ML techniques, known as neural networks, is known as a universal approximator for any relation between the “affecting parameter” and “affected parameter.” It takes raw data, trains itself, and performs. The common source of data can be previously published data by researchers, data generated from experiments, data generated from computational analysis, new observations, or a combination of two or more sources.

As the computing power of computers is advancing day by day, more complex problems with several influencing parameters can be solved in less time. The influence of ML in the domain of heat transfer discovers data-driven research and patterns. More complex problems associated with the field of bio-heat transfer, convection problems with a large number of affecting parameters, and the study of the porous medium, multi-phase flow, nanofluids, and others can be analyzed easily and effectively. In the heat transfer domain, interaction occurs within the system, as in the case of conduction, due to metallurgical conditions, some gradient, or some unforeseen reasons. Another kind of interaction, i.e., the system's interaction with the surrounding or universe (everything excluding the system and surrounding) through the boundary, as in the case of convection and radiation, can be handled easily. Any problems in the domain of conduction, convection, radiation, or a combination of these, in their most generalized form, i.e., with very few or zero assumptions, repel very far from the analytical solution. This issue is not only limited to heat transfer but to almost all real situations dealing with the areas of engineering, finance, cybercrimes, medicine, and many others. Supervised learning comes with algorithms for generating a relation between input parameters and output parameters (regression); results are the interaction between the fluids properties (thermos-physical), operating conditions, system geometry, and many others (fouling, metallurgy, surface conditions, anisotropicity) can be considered (due to increasing computational speed).

Significant deviation from the predefined performance, i.e., failure prediction, can also be analyzed. Another supervised type of ML, known as classification, can be helpful for the accurate classification of flow patterns and boiling flow regimes. It can also be analyzed for microchannels. Optimization of thermal systems can be thought of as a nonlinear type of problem. It's more like an iterative process coupled with unsupervised ML. Certainly, the image or scenario generation ability of ML is something we couldn't think of in the past; based on the collected data, it would help researchers go beyond today's discovery's current limit. Any real-world problems have several influencing parameters, which can be reduced to a great extent so that the analytical solution of the above-mentioned can be made possible. Algorithms used in any learning, either supervised or unsupervised, require data to predict or produce some kind of outcome. Still, reinforcement learning generates data for predicting or generating outcomes. In the future, researchers can use it in result generation for problems with many rules and their dependencies in complex environments—system optimization and control associated with thermal systems, which can be utilized to explore conclusive results.

4.1 Future scope associated with PINN

The physics-informed neural network represents an exciting scientific computing and machine learning frontier. It offers many future applications and directions, such as:

Multi-physics and multiphase system: This technique can be extended to handle complex physics involving multi-domain phenomena. This can be achieved by coupling different PDEs to model realistic scenarios [177].

Uncertainty quantification: Adding scenarios like uncertainty or noise into PINN improves the reliability in real-world applications [178].

Optimization and control: PINN provides a framework for optimizing the design or process for a real-world or real-time problem or situation [179].

Scope in healthcare and industry: PINNs can potentially revolutionize energy systems, biomedical, aerospace, and material science fields. It can enable predictive maintenance, personalize medicine and optimization of industrial process by integrating data-driven approach with physics principles [180].

Data-driven discovery and scientific insight: It uncovers hidden patterns in experimental data, suggests hypotheses, and aids in model

discovery by capturing complex non-linear relationships that traditional physics-based models might overlook [181].

Integration with experimental data and simulation: Integrating PINNs with experimental or simulation data allows for hybrid modeling approaches, improving model accuracy [182].

4.2 Conclusions

This work discussed a physics-informed neural network approach to solve the heat transfer PDE under specified boundaries and initial conditions. It was based on training a neural network using a total loss function defined to satisfy the PDE, IC, and BCs. Apart from this, physics-based information was expressed using the theory associated with heat transfer. For training, a non-dimensional form of the governing equation is used to address the difference in magnitudes of difference loss terms. The predictions made by trained neural networks were validated using the analytical solution.

Building a neural network based on knowledge of physics offers many advantages over analytical, numerical, experimental, or any other existing solution methods for solving PDEs or ODEs. Once a neural network is trained well, it can be used for real-time simulation. This method is in its early phase. This work considers that the time and space complexity associated with this method and optimization of the algorithm efficiency can also be improved in the coming time.

During the development of the neural network for the study of the slab, we used the exponential function and the sine function as activation functions for the neurons representing the time and spatial dimensions, respectively. We then employed the powerful Adam with a learning rate of 0.001, followed by the efficient L-BFGS, to optimize the model. The L2 relative error was accepted as the metric. This model predicts the temperature accurately up to the fourth decimal place when compared with the exact or true results, demonstrating the efficiency of our optimization methods.

In the development of a neural network for the study of fin, tanh is used as an activation function for the neurons present in the network. Adam, with a learning rate of 0.001 and L2 relative error, is assigned to the model as an optimizer and as a metric. This model predicts temperature up to the third decimal place compared to the exact temperature.

Chapter-5

5. REFERENCES

[1] S. Daniel, M. Oscar, H. David, d. F. Martín and F. Esteban, "Reinforcement learning to maximize wind turbine energy generation,," *Expert Systems with Applications,,* vol. 249, 2024.

[2] R. Federica, V. Matteo, S. Guido and F. Egidio, "Brain-inspired meta-reinforcement learning cognitive control in conflictual inhibition decision-making task for artificial agents," *Neural Networks,* vol. 152, 2022.

[3] K. Xu, W. Yujie and C. Zonghai, "Xu Kang, Yujie Wang, Zonghai Chen,A reinforcement learning energy management strategy for fuel cell hybrid electric vehicles considering driving condition classification,," *Sustainable Energy, Grids and Networks,,* vol. 38, 2024.

[4] L. Lujie, Y. Jun and Y. Bingxin, "A dynamic mission abort policy for transportation systems with stochastic dependence by deep reinforcement learning," *Reliability Engineering & System Safety.*

[5] H. Hui, G. Xiangdi, C. Yue, T. Jinrui, H. Tingting and F. Rengcun, "Model-free dynamic management strategy for low-carbon home energy based on deep reinforcement learning accommodating stochastic environments," *Energy and Buildings,* vol. 278, 2023.

[6] L. Ning, S. Martin and G. Yuchen, "A systematic review of unsupervised learning techniques for software defect prediction," *Information and Software Technology,* vol. 122, 2020.

[7] W. Roung-Shiunn and C. Po-Hsuan, "Customer segmentation of multiple category data in e-commerce using a soft-clustering approach,," *Electronic Commerce Research and Applications,,* vol. 10, pp. 331-341, 2011.

[8] K. Hamed, K. Mohammed, A. C. James, H. M.élynda, D. Yousef, U. Maiken, F. Sigfredo and G. Marek, "Advanced evaluation techniques: Gas sensor networks, machine learning, and chemometrics for fraud detection in plant and animal products," *Sensors and Actuators A: Physical,* vol. 370, 2024.

[9] A. F. Marco, E. S. Fernando, C. Carlos and C. A. Jose, "Thermographic image processing analysis in a solar concentrator with hard C-means clustering,," *Energy Reports,,* vol. 9, 2023.

[10] T. Chengyi, L. Jianhong, F. Ying and P. Xuwei, "Dimensionality reduction in stochastic complex dynamical networks," *Chaos, Solitons & Fractals.*

[11] T. Francesco, T. Qinghua, P. Panagiotis and A. S. Johan, "Deep Kernel Principal Component Analysis for multi-level feature learning," *Neural Networks,* vol. 170, 2024.

[12] C. Chiranjib, B. Manojit, P. Soumen and L. Sang-Soo, "From machine learning to deep learning: Advances of the recent data-driven paradigm shift in medicine and healthcare,," *Current Research in Biotechnology,,* vol. 7, 2024.

[13] V. Belis, O. Patrick and K. A. Thea, "Machine learning for anomaly detection in particle physics,," *Reviews in Physics,,* vol. 12, 2024.

[14] A. Tanveer, C. Huanxin, H. Ronggeng, Y. Guo, W. Jiangyu, S. Jan, M. Hafiz, A. zeem, A. H. M. Syed and K. Muhammad, "Supervised based machine learning models for short, medium and long-term energy prediction in distinct building environment," *Energy,* vol. 158, 2018.

[15] Z. Guomei, X. Tianhao, F. Xuemeng, Z. Wenlin, W. Liquan, L. Jiaping, H. Yaxi and D. Lei, "Machine-learning-assisted multiscale modeling strategy for predicting mechanical properties of carbon fiber reinforced polymers," *Composites Science and Technology,* vol. 248, 2024.

[16] H. Dennis, F. B. Stefan, W. N. Eric, K. Ajay and D. Dursun, "Machine learning in marketing: Recent progress and future research directions,," *Journal of Business Research,,* vol. 170, 2024.

[17] P. Debidutta, R. Sougata and R. Raghu, "Applications of artificial intelligence and machine learning in the financial services industry: A bibliometric review," *Heliyon,* vol. 10, 2024.

[18] L. T. Zhen, J. Liyuan, L. Nan, L. Siqi, M. Di, Z. Xiaoman, Y. N. Wei, F. T. Ting, M. L. Deborah, J. C. Kai, H. John, L. Yong, S. M. G. Rick and S. W. T. Daniel, "Federated machine learning in healthcare: A systematic review on clinical applications and technical architecture,," *Cell Reports Medicine,,* vol. 5, 2024.

[19] E. B. Khadija and S. d. S. Rafael, "Learning in Big Data: Introduction to Machine Learning," in *Knowledge Discovery in Big Data from Astronomy and Earth Observation*, 2020, pp. 225-249.

[20] C. Yatong, K. Rui and K. Linghe, "Applications of artificial intelligence in astronomical big data," in *Big Data in Astronomy,*, 2020, pp. 347-375.

[21] E. Ahmed, R. Sarah, L. Jana, B. Andrew and G. Pradeep, "Application of classification machine learning algorithms for characterizing nutrient transport in a clay plain agricultural watershed,," *Journal of Environmental Management,,* vol. 345, 2023.

[22] R. F. Peter and M. T. Brent, "A consistent framework for Horton regression statistics that leads to a modified Hack's law," *Geomorphology,* vol. 102, 2008.

[23] O. M. Marcele, L. N. Sergio, S. D. Paulo and T. Sergios, "Machine learning: Review and trends," in *Signal Processing and Machine Learning Theory*, 2024, pp. 869-959.

[24] H. Sijia, "Linear regression analysis,," in *International Encyclopedia of Education (Fourth Edition),*, 2023, pp. 548-557.

[25] Z. Tao, Z. Qingzhao and W. Qihua, "Model detection for functional polynomial regression," *Computational Statistics & Data Analysis,* vol. 70, 2014.

[26] G. Raoof and F. Nikoo, "Support Vector Machine: Principles, Parameters, and Applications," in *Handbook of Neural Computation*, 2017, pp. 515-535.

[27] Z. Zichen, H. Wei-Chiang and D. Yongquan, "Multi-hyperplane twin support vector regression guided with fuzzy clustering," *Information Sciences,* 2024.

[28] B. Abdul-Lateef and T. Abdulwaheed, "Modelling and investigating the impacts of climatic variables on ozone concentration in Malaysia using correlation analysis with random forest, decision tree regression, linear regression, and support vector regression," *Chemosphere,* 2022.

[29] S. Xuejin, J. C. O. Maria, P. A. Taranenko, K. Afrasyab, A. Mohammed and A. Anas, "Modeling and optimization of vegetable oil biodiesel production with heterogeneous nano catalytic process: Multi-layer perceptron, decision regression tree, and K-Nearest Neighbor methods," *Environmental Technology & Innovation,* vol. 27, 2022.

[30] S. Zhigang, W. Guotao, L. Pengfei, W. Hui, Z. Min and L. Xiaowen, "An improved random forest based on the classification accuracy and correlation measurement of decision trees,," *Expert Systems with Applications,* 2024.

[31] J. Mudi, W. Jiaqi, H. Lianyu and H. Zengyou, "Random forest clustering for discrete sequences,," *Pattern Recognition Letters,,* 2023.

[32] K. Kim, R. Warren, M. B. Katharine, J. M. Callum, C. T. Rachel, C. C. G. Walther, H. Crona, D. Peter and H. Paul, "Lasso penalisation identifies consistent trends over time in landscape and climate factors influencing the wintering distribution of the Eurasian Curlew (Numenius arquata)," *Ecological Informatics,* vol. 77, 2023.

[33] H. Liyang, P. Pierre and K. Jalal, "Trading data for wind power forecasting: A regression market with lasso regularization," *Electric Power Systems Research,* vol. 212, 2022.

[34] W. Xiaoyu, W. Xingyuan, M. Q. L. Bin, W. Chunpeng and S. Yunqing, "High-performance reversible data hiding based on ridge regression prediction algorithm," *Signal Processing.*

[35] j. W. Cort and M. Kenji, "Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance," *Climate Research,* vol. 30, 2005.

[36] S. Pardeep, D. Manoj and S. Sumit, "Different normalization techniques as data preprocessing for one step ahead forecasting," in *Artificial Intelligence for Renewable Energy Systems,*, 2022, pp. 209-230.

[37] E. G.-G. Rubén, A.-M. Gerardo, V. Héctor, P. A. J. R. Sylvia and D. Reginaldo, "Predictive performance from abundance distribution models of Vinciguerria lucetia larvae in the southern portion of the California current system using XGBOOST," *Deep Sea Research Part II: Topical Studies in Oceanography,* vol. 212, 2023.

[38] M. R. Sheldon, "Regression," *Introduction to Probability and Statistics for Engineers and Scientists (Fifth Edition),* Vols. 357-444, 2014.

[39] T. Kanishka, R. Chinmay and M. M. Harshvardhan, "Regression analysis," in *Artificial Intelligence and Machine Learning for EDGE Computing,*, 2022, pp. 53-63.

[40] T. Sergios, "Mean-Square Error Linear Estimation," in *Machine Learning (Second Edition)*, 2020, pp. 121-177.

[41] I. Casas, "Neural Networks," in *International Encyclopedia of Human Geography (Second Edition),*, 2020.

[42] T. Edmondo, "Networks with trainable amplitude of activation functions," *Neural Networks,* pp. 471-493, 2001.

[43] R. D. Shiv, K. S. Satish and B. C. Bidyut, "Activation functions in deep learning: A comprehensive survey and benchmark," *Neurocomputing,* pp. 92-108, 2022.

[44] B. Wadii, A. Eman, A. Ayyub, K. Anis and M. Nabil, "An effective weight initialization method for deep learning: Application to satellite image classification," *Expert Systems with Applications,* 2024.

[45] N. Viet-Ha, H. Nhat-Duc, N. Hieu, T. T. N. Phuong, T. B. Tinh, V. H. Pham, S. Pijush and T. B. Dieu, "Effectiveness assessment of Keras based deep learning with different robust optimization algorithms for shallow landslide susceptibility mapping at tropical area," *CATENA,* 2020.

[46] D. Thijs, B. Bert and C. Jan, "CFD simulation of heat transfer at surfaces of bluff bodies in turbulent boundary layers: Evaluation of a forced-convective temperature wall function for mixed convection," *Journal of Wind Engineering and Industrial Aerodynamics,* pp. 439-446, 2012.

[47] A. Ozgen, B. Ç. Andaç, C. Muhammet, K. Yakup and S. D. Ahmet, "Machine learning approach to predict the heat transfer coefficients pertaining to a radiant cooling system coupled with mixed and forced convection,," *International Journal of Thermal Sciences,,* vol. 178, 2022.

[48] A. Ozgen, Ç. Alican, S. D. Ahmet, K. Aliihsan, Ç. Gürsel, G. Zafer and W. Somchai, "A novel ANN-based approach to estimate heat transfer coefficients in radiant wall heating systems,," *Energy and Buildings,,* 2017.

[49] V. Vikas, N. Ratnadeep and T. Rahul, "Heat transfer prediction for radiant floor heating/cooling systems using artificial neural network (ANN)," *Heat transfer,* vol. 52, pp. 3135-3152, 2023.

[50] B. Ç. Andaç, A. Ozgen, K. Yakup, K. Aliihsan and S. D. Ahmet, "Experimental and numerical investigations on the heat transfer characteristics of a real-sized radiant cooled wall system supported by machine learning,," *International Journal of Thermal Sciences,,* vol. 191, 2023.

[51] X. Qian, W. Yifan, L. Xiaoqiang, Y. Zhao, L. Jiali, X. Zhihong, W. Xianxuan, C. Jiajia and X. Qiyan, "Two-dimensional transient heat transfer model of moving quenching jet based on machine learning," *International Journal of Heat and Mass Transfer,,* vol. 191, 2022.

[52] L.-B. Alejandro, I.-G. Fernando, M. C.-I. José and R. G.-C. José, "GMDH ANN to optimise model development: Prediction of the pressure drop and the heat transfer coefficient during condensation within mini-channels,," *Applied Thermal Engineering,* 2018.

[53] B. Suvanjan, K. V. Devendra, C. Shramona, R. Rahul, A. Issakhov and S. Mohsen, "Turbulent Flow Heat Transfer through a Circular Tube with Novel Hybrid Grooved Tape Inserts: Thermohydraulic Analysis and Prediction by Applying Machine Learning Model," *Sustainability,* 2021.

[54] S. S and S. M, "Employing ensemble machine learning techniques for predicting the thermohydraulic performance of double pipe heat exchanger with and without turbulators,," *Thermal Science and Engineering Progress,,* vol. 47, 2024.

[55] C. Nevin, T. Beyda, K. Sinan and T. Vedat, "Performance optimization of a heat exchanger with coiled-wire turbulator insert by using various machine learning methods,," *International Journal of Thermal Sciences,,* vol. 192, 2023.

[56] T. H. Matthew, M. F. Brian and G. Srinivas, "Universal condensation heat transfer and pressure drop model and the role of machine learning techniques to improve predictive capabilities,," *International Journal of Heat and Mass Transfer,,* 2021.

[57] R. S. Sandip Dutta, "Nonlinear Optimization of Turbine Conjugate Heat Transfer with Iterative Machine Learning and Training Sample Replacement," *Modelling of Thermal and Energy Systems,* 2020.

[58] M. S. Kang, S. G. Park and C. T. Din, "Heat transfer enhancement by a pair of asymmetric flexible vortex generators and thermal performance

prediction using machine learning algorithms,," *International Journal of Heat and Mass Transfer,,* vol. 200, 2023.

[59] D. Manish, S. P. Om and R. Nirupam, "Flow boiling heat transfer analysis of Al2O3 and TiO2 nanofluids in horizontal tube using artificial neural network (ANN)," *Journal of Thermal Analysis and Calorimetry,* 2019.

[60] B. Alireza, K. Mostafa, A. N. Mohammad, H. A. Mohammad and Y. Wei-Mon, "Sensitivity analysis and application of machine learning methods to predict the heat transfer performance of CNT/water nanofluid flows through coils,," *International Journal of Heat and Mass Transfer,,* 2019.

[61] C. Haoran, C. Boyang, X. Congyi, M. Zhaoming, B. Haozhi and D. Ming, "Prediction of heat transfer coefficients for steam condensation in the presence of air based on ANN method,," *International Journal of Advanced Nuclear Reactor Design and Technology,,* vol. 5, pp. 77-85, 2023.

[62] N. Elboughdiri, S. Q. Salih, B. S. Chauhan, A. Albani, T. U. K. Nutakki, F. Alturise, S. Alkhalaf and S. M, "Arc-curved microchannels engraved on segmented circular heat sink for heat transfer augmentation; ANN-based performance optimization,," *Case Studies in Thermal Engineering,,* vol. 53, 2024.

[63] K. Gaurav, T. Sulekh and K. Rajesh, "Numerical heat transfer analysis & predicting thermal performance of fins for a novel heat exchanger using machine learning,," *Case Studies in Thermal Engineering,,* vol. 21, 2020.

[64] A. L. Giovanni, M. Simone, R. Giulia, Z. Claudio, O. Ludovico and Z. Mauro, "Application of an Artificial Neural Network (ANN) for predicting low-GWP refrigerant boiling heat transfer inside Brazed Plate Heat Exchangers (BPHE),," *International Journal of Heat and Mass Transfer,,* vol. 160, 2020.

[65] N. Kedam, D. A. Uglanov, E. V. Blagin, A. A. Gorshkalev, R. A. Panshin and J. Liu, "Unified ANN model for heat transfer factor (j) and friction factor (f) prediction in offset strip and wavy fin PFHEs,," *Case Studies in Thermal Engineering,,* vol. 53, 2024.

[66] B. Ç. Andaç, A. Dogan, M. Hatice, S. D. Ahmet and W. Somchai, "Estimation of heat transfer parameters of shell and helically coiled tube heat exchangers by machine learning,," *Case Studies in Thermal Engineering,,* vol. 42, 2023.

[67] K. Christopher and M. Issam, "Review of flow boiling and critical heat flux in microgravity," *International Journal of Heat and Mass Transfer,* vol. 80, pp. 469-493, 2015.

[68] "Recapturing a Future for Space Exploration: Life and Physical Sciences Research for a New Era".

[69] C. Francis and J. Jitendra, "Workshop on Critical Issues in Microgravity Fluids, Transport, and Reaction Proces".

[70] T. Shaghayegh, S. Behrang, M. A. and B. Akhavan, "Prediction of heat transfer coefficient and pressure drop of R1234yf and R134a flow condensation in horizontal and inclined tubes using machine learning techniques," *International Journal o Re rigeration,* vol. 152, 2023.

[71] Z. Guangya, L. Shirui, Z. Dalin, C. Weijian, L. Jie and W. Tao, "Transfer learning model to predict flow boiling heat transfer coefficient in mini channels with micro pin fins,," *International Journal of Heat and Mass Transfer,* vol. 2024, 220.

[72] Q. Yue, V. Tinh, G. Deepak, L. Hyounsoon and R. K. Chirag, "A systematic approach to optimization of ANN model parameters to predict flow boiling heat transfer coefficient in mini/micro-channel heatsinks,," *International Journal of Heat and Mass Transfer,,* 2023.

[73] H. S. J. Luiz, R. B. Jader and K. d. S. Alexandre, "Multi-parameter classification and quantification of R-134a condensation using machine learning," *Applied Thermal Engineering,* vol. 231, 2023.

[74] Z. Guangya, W. Tao and Z. Dalin, "Machine learning based approach for the prediction of flow boiling/condensation heat transfer performance in mini channels with serrated fins,," *International Journal of Heat and Mass Transfer,,* vol. 166, 2021.

[75] H. Yichuan, H. Chengzhi, J. Bo, S. Zhehao, M. Jing, L. Hongyang and T. Dawei, "Data-driven approach to predict the flow boiling heat transfer coefficient of liquid hydrogen aviation fuel,," *Fuel,* vol. 324, 2022.

[76] Z. Liwei, G. Deepak, Q. Yue, K. Sung-Min, M. Issam and R. K. Chirag, "Machine learning algorithms to predict flow condensation heat transfer coefficient in mini/micro-channel utilizing universal data,," *International Journal of Heat and Mass Transfer,,* vol. 162, 2020.

[77] T. Jiguo, Y. Shengzhi, M. Chen, L. Hongtao and M. Zhengyu, "Prediction of heat transfer of bubble condensation in subcooled liquid using machine learning methods," *Chemical Engineering Science,,* vol. 271, 2023.

[78] K. Siavash, F. R. Kazi, S. Youngjoon, W. Yoonjin and M. Nenad, "Machine learning enabled condensation heat transfer measurement," *International Journal of Heat and Mass Transfer,,* vol. 192, 2022.

[79] B. Ari, Q. Yue, R. K. Chirag and F. Roger, "Consolidated modeling and prediction of heat transfer coefficients for saturated flow boiling in mini/micro-channels using machine learning methods,," *Applied Thermal Engineering,,* vol. 210, 2022.

[80] Y. Bin, Z. Xin, W. Boan, L. Minzhang, L. Yifan, L. Zhihan and W. Faming, "Computer Vision and Machine Learning Methods for Heat Transfer and

Fluid Flow in Complex Structural Microchannels: A Review," *energies,* 2023.

[81] R. T. Al-Jarrah, "A novel machine-learning schemes to predict heat transfer coefficient during condensation of CO2 in porous media," *Journal of Thermal Analysis and Calorimetry,* 2023.

[82] Hongwang Zhao, "Computational modeling of nanofluid heat transfer using Fuzzy-based bee algorithm and machine learning method,," *Case Studies in Thermal Engineering,,* no. 2024.

[83] D. Disha, R. Harish, K. Rajiv and M. R, "CFD and Machine Learning based Simulation of Flow and Heat Transfer Characteristics of Micro Lattice Structures," *IOP Conference Series: Earth and Environmental Science,* 2021.

[84] D. M. Mahsa, A. Nima, T. Alireza, M. Milad and M. Amirhosein, "Comparative Analysis of Machine Learning and Numerical Modeling for Combined Heat Transfer in Polymethylmethacrylate," *polymers,* 2022.

[85] A. Rasool, M. N. A. Javad, A. Abolhasan, R. M. A. M. Mohammad, Z. Dan and K. Nader, "A machine learning approach to the prediction of transport and thermodynamic processes in multiphysics systems - heat transfer in a hybrid nanofluid flow in porous media,," *Journal of the Taiwan Institute of Chemical Engineers,,* vol. 124, pp. 290-306, 2021.

[86] Z. Xiao, Y. Ruomiao, W. Qifan, Y. Yuchao, Z. Yu, F. Jiahong and L. Zhentao, "Comparison of GRNN and RF algorithms for predicting heat transfer coefficient in heat exchange channels with bulges," *Applied Thermal Engineering,,* vol. 217, 2022.

[87] D. Martin, T. Adrien and B. Françoise, "Numerical development of heat transfer correlation in asymmetrically heated turbulent channel flow," *International Journal of Heat and Mass Transfer,* vol. 164, 2021.

[88] N. Feng, W. Haocheng, Z. Yanxing, S. Qinglu, Y. Shiqi and G. Maoqiong, "A universal correlation for flow condensation heat transfer in horizontal tubes based on machine learning,," *International Journal of Thermal Sciences,,* vol. 184, 2023.

[89] S. A. Maryam, M. Hedayati, A. Ahmad and S. Mehdi, "Comprehensive heat transfer correlation for water/ethylene glycol-based graphene (nitrogen-doped graphene) nanofluids derived by artificial neural network (ANN) and adaptive neuro-fuzzy inference system (ANFIS)," *Heat Mass Transfer,* vol. 3073–3083, p. 3073–3083.

[90] Z. Chong, S. Yunren and W. Wei, "Machine learning-assisted correlations of heat/mass transfer and pressure drop of microchannel membrane-based desorber/absorber for compact absorption cycles,," *International Journal of Heat and Mass Transfer,,* vol. 214, 2023.

[91] T. Jiguo, Y. Shengzhi, Hongtao and Liu, "Development of correlations for steam condensation over a vertical tube in the presence of noncondensable gas using machine learning approach,," *International Journal of Heat and Mass Transfer,,* vol. 201, 2023.

[92] G.-U. Alexander, F. Kazuki, O. Hajime and F. Yasuhiro, "Deriving local Nusselt number correlations for heat transfer of nanofluids by genetic programming,," *International Journal of Thermal Sciences,,* 2023.

[93] K. Beomjin, E. Faizan and K. H. Leslie, "Machine learning for heat transfer correlations," *International Communications in Heat and Mass Transfer,* p. 116, 2020.

[94] T. V. Anh, G. Shrey, V. Paul-Alexander, G. Tim and B. Thomas, "Machine learning-based predictive modeling of contact heat transfer," *International Journal of Heat and Mass Transfer,,* vol. 174, 2021.

[95] R. Zhang, S. Qing, X. Zhang, J. Li, Y. Liu and X. Wen, "Experimental investigation and machine learning modeling of heat transfer characteristics for water based nanofluids containing magnetic Fe3O4 nanoparticles,," *Materials Today Communications,,* vol. 36, 2023.

[96] C. Yuli, X. Ruina and P. Jiang, "Physics-informed machine learning based RANS turbulence modeling convection heat transfer of supercritical pressure fluid,," *International Journal of Heat and Mass Transfer,* vol. 201, 2023.

[97] H. L. Dong, M. Y. Jee, Y. K. Hui, J. H. Dong, J. Y. Byong and J. J. Jae, "Application of the machine learning technique for the development of a condensation heat transfer model for a passive containment cooling system,," *Nuclear Engineering and Technology,,* vol. 54, 2022.

[98] T. H. Matthew, M. C. Sarah and G. Srinivas, "Machine-learning-based heat transfer and pressure drop model for internal flow condensation of binary mixtures," *International Journal of Heat and Mass Transfer,* 2022.

[99] M. Mohammed, M. E.-S. Emad, O. A. Ahmad, A. Rolla, A. E. Mohamed, M. E. Samir and H. E. Ammar, "Modelling of thermo-hydraulic behavior of a helical heat exchanger using machine learning model and fire hawk optimizer," *Case Studies in Thermal Engineering,* vol. 49, 2023.

[100] Sadegh Motahar, "Experimental study and ANN-based prediction of melting heat transfer in a uniform heat flux PCM enclosure,," *Journal of Energy Storage,,* vol. 30, 2020.

[101] N. A. K. Muhammad, A. A. Usman Ghafoor, U. H. Zeeshan, U. Hafeez, H. G. Iftikhar and M. Asghari, "Prediction of thermal diffusivity of volcanic rocks using machine learning and genetic algorithm hybrid strategy,," *International Journal of Thermal Sciences,,* vol. 192, 2023.

[102] L. Kaiqi, H. Robert and H. Hailong, "Application of machine learning algorithms to model soil thermal diffusivity,," *International Communications in Heat and Mass Transfer,,* vol. 149, 2023.

[103] Z. Ruizeng, T. Wentao, X. Siyuan, Q. Qinggang and Z. Xiaojing, "ANN model with feature selection to predict turbulent heat transfer characteristics of supercritical fluids: Take CO2 and H2O as examples,," *International Journal of Thermal Sciences,,* 2023.

[104] D. Ma, Y. Li, T. Zhou and Y. Huang, "Research on prediction and analysis of supercritical water heat transfer coefficient based on support vector machine," *Nuclear Engineering and Technology,* vol. 55, 2023.

[105] A. Nurlaily, N. H. Hieu, O. Jong-Taek and M. Normah, "Application of machine learning to the prediction of the boiling heat transfer coefficient of R32 inside a multiport mini-channel tube," *Journal of Thermal Analysis and Calorimetry,* vol. 148, p. 3137–3153, 2022.

[106] T. Iman, G. Ashkan, A. Ehsanolah and M. Mojtaba, "The effects of acoustic pressure level on the heat transfer performance of a wavy channel: Calibrated RANS model with machine learning application,," *Engineering Analysis with Boundary Elements,,* vol. 146, 2023.

[107] Y. Liang, Y. Haonan and J. Yonglin, "Review on the key technologies and future development of insulation structure for liquid hydrogen storage tanks," *International Journal of Hydrogen Energy,* vol. 57, 2024.

[108] Y. Huan, W. Jiarui, W. Jian and X. Haolin, "Assessment of machine learning models and conventional correlations for predicting heat transfer coefficient of liquid hydrogen during flow boiling,," *International Journal of Hydrogen Energy,,* vol. 49, 2024.

[109] A. M. Ahmad and M. A. Hafiz, "Batteries temperature prediction and thermal management using machine learning: An overview,," *Energy Reports,,* vol. 10, 2023.

[110] N. Julius, L. Seppo, V. Ville-Valtteri and A. Tuomas, "Coupling of Solidification and Heat Transfer Simulations with Interpretable Machine Learning Algorithms to Predict Transverse Cracks in Continuous Casting of Steel," *Steel Research International,* 2024.

[111] K. Vishnu, P. Dhiraj, R. S. Vansh, K. Rishika, G. Shaifu and P. Harish, "Prediction of CHF location through applied machine learning," *Progress in Nuclear Energy,* vol. 169, 2024.

[112] M. Issam, J. D. Steven and S. D. V, "Prediction technique for flow boiling heat transfer and critical heat flux in both microgravity and Earth gravity via artificial neural networks (ANNs),," *International Journal of Heat and Mass Transfer,,* vol. 220, 2024.

[113] B. Ankan, S. Aritra, B. Sumanta, c. R. Prokash and K. Balaram, "A Review of Artificial Intelligence Methods in Predicting Thermophysical Properties of Nanofluids for Heat Transfer Applications," *Energies,* vol. 17, 2024.

[114] B. Mehdi and H. Saeed, "Electronics cooling with nanofluids: A critical review," *Energy Conversion and Management,* vol. 172, 2018.

[115] M. Nandyala, S. Neetu and T. Prabal, "A mathematical model for understanding nanoparticle biodistribution after intratumoral injection in cancer tumors," *Journal of Drug Delivery Science and Technology,* 2022.

[116] R. G. Sreedhara, M. J. S. S. Victor, V. M. P and V. C. B, "Numerical study of blood perfusion and nanoparticle transport in prostate and muscle tumours during intravenous magnetic hyperthermia," *Alexandria Engineering Journal,* vol. 60, 2021.

[117] J. Shaik, V. E. Sathishkumar and R. R. R. Seethi, "Numerical and Machine Learning Approach for Fe3O4-Au/Blood Hybrid Nanofluid Flow in a Melting/Non-Melting Heat Transfer Surface with Entropy Generation," *symmetry,* 2023.

[118] U. Atta, K. Mustafa, H. Ghulam, S. Mahir, Z. K. Rehan and S. Khairuddin, "Reliable prediction of thermophysical properties of nanofluids for enhanced heat transfer in process industry: a perspective on bridging the gap between experiments, CFD and machine learning," *Journal of Thermal Analysis and Calorimetry,* 2023.

[119] R. Maziar, P. Paris and E. K. George, "Inferring solutions of differential equations using noisy multi-fidelity data," *Journal of Computational Physics,* vol. 335, pp. 736-746, 2017.

[120] R. Maziar, P. Paris and E. K. George, "Machine learning of linear differential equations using Gaussian processes," *Journal of Computational Physics,* vol. 348, pp. 683-693.

[121] H. Owhadi, "Bayesian Numerical Homogenization," *Multiscale Modeling & Simulation,* vol. 13, pp. 812-828, 2015.

[122] C. E. Rasmussen, "Gaussian processes in machine learning," *Summer school on machine learning,* pp. 63-71, 2003.

[123] G. B. Atılım, A. P. Barak, A. R. Alexey and M. S. Jeffrey, "Automatic differentiation in machine learning: a survey," *The Journal of Machine Learning Research,* vol. 18, 2017.

[124] K. George and S. Spencer, Spectral/hp Element Methods for Computational Fluid Dynamics, 2005.

[125] D. Vikas and S. Balaji, "Physics Informed Extreme Learning Machine (PIELM)–A rapid method for the numerical solution of partial differential equations," *Neurocomputing,* 2019.

[126] E. L. Isaac, L. Aristidis and I. F. Dimitrios, "Artificial Neural Networks for Solving Ordinary and Partial Differential Equations," *IEEE TRANSACTIONS ON NEURAL NETWORKS,* vol. 9, 1998.

[127] C. Samarjeet, B. C, P. V. S and R. Y. Govinda, "Estimation of principal thermal conductivities of layered honeycomb composites using ANN–GA based inverse technique," *International Journal of Thermal Sciences,* 2017.

[128] H. Kurt, S. Maxwell and W. Halbert, "Multilayer feedforward networks are universal approximators," *Neural Networks,* vol. 2, no. 5, pp. 359-366, 1989.

[129] G. B. Atilim, A. P. Barak, A. R. Alexey and M. S. Jeffrey, "Auomatic differentiation in matchine learning: a survey," 2018.

[130] H. S. Carslaw and J. C. Jaeger, "Some problems in the mathematical theory of the conduction of heat," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science,* 1938.

[131] W. H. David and M. N. Özişik, Heat conduction, 2012.

[132] M. N. Özışık:, Heat Transfer: A basic approach, New York, 1985.

[133] W. M. Kays and A. L. London, Compact Heat Exchangers, Krieger Publishing Company, 1998.

[134] B. A. Boley and J. H. W. , Theory of Thermal Stresses, Dover Publications, 1960.

[135] Y. A. Cengel, Heat Transfer: A Practical Approach, McGraw-Hill, 2003.

[136] F. P. Incorpera and T. L. D. P Bergman, Fundamentals of Heat and Mass Transfer, John Wiley & Sons, 2006.

[137] A. Bejan, Convection Heat Transfer, John Wiley & Sons, 2013.

[138] H. L. S. Kakac, Heat Exchangers: Selection, Rating, and Thermal Design, CRC Press, 2002.

[139] V. P. Carey, Liquid-Vapor Phase-Change Phenomena: An Introduction to the Thermophysics of Vaporization and Condensation Processes in Heat Transfer Equipment., Taylor & Francis, 1992.

[140] F. Kreith and D. Y. Goswami, Handbook of Energy Efficiency and Renewable Energy, CRC press, 2007.

[141] H. S. Carslaw and J. C. Jaeger, Conduction of Heat in Solids, Oxford University Press, 1959.

[142] S. V. Patankar, Numerical Heat Transfer and Fluid Flow, Hemisphere Publishing Corporation, 1980.

[143] O. C. Zienkiewicz and R. L. Tylor, The Finite Element Method: Volume 1, The Basis. Butterworth-Heinemann, 2000.

[144] H. K. Versteeg and W. Malalasekera, An Introduction to Computational Fluid Dynamics: The Finite Volume Method, Pearson Education, 2007.

[145] J. P. Holman, Experimental Methods for Engineers, McGraw-Hill Education, 2010.

[146] E. R. G. Eckert and R. J. Goldstein, Measurements in Heat Transfer, Hemisphere Publishing Corporation, 1976.

[147] E. Buckingham, "On Physically Similar Systems; Illustrations of the Use of Dimensional Equations," *Physical Review,* vol. 4, pp. 345-376, 1914.

[148] F. M. White, Fluid Mechanics, McGraw-Hill Education, 2006.

[149] W. M. Rohsenow and Y. I. C. J. P Hartnett, Handbook of Heat Transfer, MccGraw-Hill Education., 1998.

[150] W. M. Kays and M. E. Crawford, Convective Heat and Mass Transfer, cGraw-Hill Education, 1993.

[151] A. D. Kraus and A. B. Cohen, Design and Analysis of Heat Sinks, Wiley-Interscience, 1995.

[152] G. E. Karniadakis, I. G. Kevrekidis, S. W. P. Perdikaris and L. Yang, "Physics-informed machine learning," *Nature Reviews Physics,* vol. 3, no. 422-440, 2021.

[153] S. L. Brunton, B. R. Noack and P. Koumoutsakos, "Machine learning for fluid mechanics," *Annual Review of Fluid Mechanics,* vol. 52, pp. 477-508, 2020.

[154] Y. Liu, S. Ramachandran and M. Baldea, "Data-driven modeling and predictive control in process systems engineering: Advances and challenges," *Computers & Chemical Engineering,* vol. 136, 2020.

[155] M. Raissi, P. Perdikaris and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational Physics,* vol. 378, pp. 686-707, 2019.

[156] Z. Yang and P. Perdikaris, "Adversarial uncertainty quantification in physics-informed neural networks," *Journal of Computational Physics,* vol. 394, pp. 136-152.

[157] N. Muralidhar, C. R. Harris and A. D. White, "Network models for predicting molecular properties," *Journal of Chemical Information and Modeling,* vol. 59, pp. 3615-3621, 2019.

[158] S. Bhatnagar, S. P. Y. Afshar, D. K and K. S, "Prediction of aerodynamic flow fields using convolutional neural networks," *Computational Mechanics,* vol. 64, pp. 525-545, 2019.

[159] L. Zhang, L. Lu and G. E. Karniadakis, "Physics-informed neural networks for multi-physics and multi-scale problems," *Journal of Computational Physics,* vol. 393, 2020.

[160] M. Raissi, P. Perdikaris and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational Physics,* vol. 378, pp. 686-707, 2018.

[161] A. Chaudhuri and S. Narasimhan, "Physics-informed deep learning for dynamic data-driven applications: A survey on promise and progress," *Journal of Computational Science,* vol. 44, 2020.

[162] A. Tripathi, "hysics-informed machine learning for real-time predictive maintenance: A case study in wind energy systems," *Applied Energy,* vol. 262, 2020.

[163] S. Pant, "Data-driven discovery of models for biological systems using neural networks.," *Computational Biology and Chemistry,* vol. 85, 2020.

[164] Z. Yang, "Physics-informed machine learning for surrogate modeling: Application to solar-thermal energy prediction," *Applied Energy,* vol. 281, 2021.

[165] J. B. Will, N. P. Kruyt and C. H. Venner, "An experimental study of forced convective heat transfer from smooth, solid spheres," *International Journal of Heat and Mass Transfer,* vol. 109, pp. 1059-1067, 2017.

[166] A. Larra˜naga, J. Martínez, J. L. Míguez and J. Porteiro, "Robust optimization of heat-transfer-enhancing microtextured surfaces based on machine learning surrogate models," *International Communications in Heat and Mass Trans□er,* vol. 151, 2024.

[167] Y. Wang, Y. Ma and H. Chao, "Machine learning and computational fluid dynamics based optimization of finned heat pipe radiator performance,," *Journal of Building Engineering,,* vol. 78, 2023.

[168] S. Chunyu, Z. M. Y. Yuhao and Z. Liu, "Arrangement optimization of spherical dimples inside tubes based on machine learning for realizing the optimal flow pattern,," *Thermal Science and Engineering Progress,,* vol. 44, 2023.

[169] W. Shizheng, D. Chunzhuo and L. Xianglei, "A machine learning strategy for modeling and optimal design of near-field radiation," *Applied Physics Letters,* 2022.

[170] S. ChunYu, Y. MinJie, L. Wei and L. ZhiChun, "Shape optimization of corrugated tube using B-spline curve for convective heat transfer enhancement based on machine learning," *Science China Technological Sciences,* 2022.

[171] N. V. Tikendra, N. Prerana, V. S. Dheerendra, S. S. Thokchom and P. Deepali, "ANN: Prediction of an experimental heat transfer analysis of concentric tube heat exchanger with corrugated inner tubes,," *Applied Thermal Engineering,,* vol. 120, pp. 219-227, 2017.

[172] M. A. Moradkhani, S. H. Hosseini and M. Karami, "Forecasting of saturated boiling heat transfer inside smooth helically coiled tubes using conventional and machine learning techniques,," *International Journal of Refrigeration,,* vol. 143, pp. 78-93, 2022.

[173] A. P. Koroleva, N. V. Kuzmenkov and M. S. Frantcuzov, "Application of machine learning methods for investigating the heat transfer enhancement performance in a circular tube with artificial roughness," *Journal of Physics: Conference Series,* 2020.

[174] P. Naphon, S. Wiriyasart, T. Arisariyawong and L. Nakharintr, "ANN, numerical and experimental analysis on the jet impingement nanofluids flow and heat transfer characteristics in the micro-channel heat sink,," *International Journal of Heat and Mass Transfer,,* vol. 131, 2019.

[175] N. V. Kuzmenkov, M. S. Frantcuzov and A. P. Koroleva, "Performance of CFD and ANN modeling of heat transfer enhancement in a circular tube with artificial roughness," *Journal of Physics: Conference Series,* 2021.

[176] S. S. Khaleduzzaman, M. R. Sohel, R. Saidur and J. Selvaraj, "Stability of Al2O3-water Nanofluid for Electronics Cooling System," *Procedia Engineering,* vol. 105, 2015.

[177] A. Alklaibi, V. V. Kotturu, C. Mouli and L. S. Sundar, "Heat transfer, and friction factor of Fe3O4–SiO2/Water hybrid nanofluids in a plate heat exchanger: Experimental and ANN predictions,," *International Journal of Thermal Sciences,* vol. 195, 2024.

[178] A. C. Y and A. J. Ghajar, Heat and mass transfer: Fundamentals and applications, 2014.

[179] P. Díaz-Morales, A. Corrochano, M. López-Martín and S. L. Clainche, "Deep learning combined with singular value decomposition to reconstruct databases in fluid dynamics," *Expert Systems with Applications,* vol. 238, 2024.

[180] R. L. Sapra, "Using R2 with caution," *Current Medicine Research and Practice,* vol. 4, 2014.

[181] T. H. Colburn and A. P. Chilton, "Mass Transfer (Absorption) Coefficients Prediction from Data on Heat Transfer and Fluid Friction," *Industrial & Engineering Chemistry,* vol. 11, pp. 1183-1187, 1934.

[182] C. Balachander, S. Arunkumar and M. Venkatesan, "Computational heat transfer analysis and combined ANN–GA optimization of hollow cylindrical pin fin on a vertical base plate," *Sadhana,* 2015.

[183] J. R. Babu and S. Gopalakrishanan, "Thermal diffusion in discontinuous media: A hybrid peridynamics-based machine learning model,," *Computers & Structures,,* vol. 290, 2024.