

Artificial Neural Network Simulation of Convective Heat Transfer around a Cylinder



By:

Hameedullah Khan

Reg. No. 956-FBAS/MSMA/F23

**Department of Mathematics and Statistics
Faculty of Sciences
International Islamic University, Islamabad
Pakistan
2025**

Artificial Neural Network Simulation of Convective Heat Transfer around a Cylinder



By:

Hameedullah Khan

Reg. No. 956-FBAS/MSMA/F23

Supervised By:

Prof. Dr. Ahmad Zeeshan

**Department of Mathematics and Statistics
Faculty of Sciences
International Islamic University, Islamabad
Pakistan
2025**

Artificial Neural Network Simulation of Convective Heat Transfer around a Cylinder

By:

Mr. Hameedullah Khan

Reg. No. 956-FBAS/MSMA/F23

A Dissertation

*Submitted in the Partial Fulfillment
of the Requirements for the Degree of
MASTER OF SCIENCE*

In

MATHEMATICS

Supervised By:

Prof. Dr. Ahmad Zeeshan

**Department of Mathematics and Statistics
Faculty of Sciences
International Islamic University, Islamabad
Pakistan
2025**

DEDICATION

I express my heartfelt gratitude and
utmost respect as I dedicated this work
to my beloved

Hazrat Muhammad (S.A.W),

My esteemed teachers

and

My dear Parents.

They have been a constant source of inspiration and
encouragement, guiding me towards greater ideals in life.

DECLARATION

I hereby declare that the work presented in this thesis is my own effort, except where otherwise acknowledged, and that the thesis is my own composition. No part of the thesis has been previously presented for any other degree.

Date _____

Mr. Hameedullah khan

MS in Mathematics

956-FBAS/MSMA/F23

ACKNOWLEDGEMENTS

In the name of Allah, the most gracious and the most merciful. First and foremost, I am thankful to Almighty **ALLAH** for guiding us on the path of knowledge after creating us as human beings and giving me the strength, knowledge and ability of this study.

Secondly, I would like to thank my respected teacher and supervisor, **Prof. Dr. Ahmad Zeeshan** whose worthy guidance and professional attitude are appreciated in completing this dissertation. I am also thankful to **Prof. Dr. Nasir Ali**, chairperson, department of Mathematics and Statistics, IIUI, for providing facilities to carry out this research work.

I am extremely grateful to my advisors **Prof. Dr. Zaheer Asghar** and **Dr. Muhammad Imran** for their support throughout this process. I thank to all my lab colleagues, fellows and friends who deserve formal acknowledgement but could not be named individually.

I would like to sincerely thank the **HEC Pakistan** for financially supporting my MS research through the funded project no: 15055 entitled “**Generalization of the Smith and Spalding Integral Method in Boundary Layer Flow with Zero Pressure Gradient.**” I worked as a Research Assistant on this project in the Department of Mathematics and Statistics, IIUI. I am especially grateful to **Prof. Dr. Ahmer Mehmood**, the Principal Investigator of the project, and **Dr. Muhammad Awais**, my mentor, for their valuable guidance and support throughout the research project.

Moreover I am extremely grateful to **My Parents** and **Siblings** for their support, encouragement, and genuine interest in my academic success. I am also thankful to **My Father** for his moral support.

(Hameed Ullah Kahn)
Re. No. 956-FBAS/MSMA/F23

PREFACE

Artificial Neural Networks (ANN), are essential for modeling and forecasting complex fluid processes in fluid dynamics. ANNs allow for the rapid analysis of large datasets from simulations or experiments, revealing details regarding turbulence, flow patterns, and heat transport. This approach is beneficial for aerospace and environmental engineering. Non-Newtonian fluids have substantial importance in engineering and manufacturing industries because of the varying nature of viscosity, which changes with applied forces. Non-Newtonian fluids are used in the manufacturing of many things, including magneto rheological dampers and brakes, body armors, protective equipment, pipelines, printer inks, and safety gears, and have many other applications. The Casson viscoelastic fluid model is a non-Newtonian fluid model that deals with the fluid flow with yield stress. Non-Newtonian fluids are classified according to how they react to changes in shear stress or shear rate. Viscoplastic fluids are a type of non-Newtonian fluid that behaves like a solid under a certain amount of stress but flows like a liquid when the stress is exceeded. Heat transfer in the boundary layer flow of a semi-infinite vertical plate with a slip boundary was applied to examine the rheological behavior of the Casson fluid [1]. They reported that increasing the value of the slip parameter reduced the velocity and temperature. Prasad et.al. [2] addressed the non-similar solution of the MHD momentum boundary layer flow of a non-Newtonian nanofluid over a circular cylinder with a non-Darcy medium. The governing equations were solved using the Keller Box Method (KBM). Ghaffar et.al. [3] Examined heat transfer of flow over cylinder using a tangent hyperbolic non-Newtonian fluid. A numerical analysis of convective transport in a vertical channel using a Casson ternary hybrid nanofluid was discussed by Yasir et al. [4]. The non-Newtonian fluid behavior in boundary layer flow and heat transfer were examined [5-8]. Mishra and Chaudhuri

[9] investigated the use of an artificial neural network and genetic algorithm. Using ANNs modeling and an experimental study, Yadav et al. [10] examined the insight flow properties of concentrated MWCNT in a water-base fluid. In order to address the properties of Hall current on MHD flow with Jeffery fluid towards a nonlinear stretchy sheet with thickness fluctuation, Awais et al. [11] looked at the artificial neural network-based solution methodology. Tian et al. [12] used hybrid machine learning techniques in conjunction with computational fluid dynamics to study the prediction of permeability in porous media. In their study, Tizakast et al. [13] investigated machine learning-based methods for modeling the movement of natural convection fluids and the movement of mass and heat in rectangular cavities containing non-Newtonian fluids. Machine-learning techniques for fluid flows at the nanoscale were studied [14-17]. Researchers employ physics-informed neural networks and machine learning techniques in boundary layer flow and heat transfer. This approach will provide the classical NN with additional physics-related information. These methods are sometimes referred to as Physics Informed Neural Networks (PINN) when used to simulate engineering and physical systems that are described by differential equations. Cuomo et al. [18] addressed the predicted solution of initial value problem with an approximation of PINN. With the advancement of technologies and use of machine learning approach PINN was used by Hubert Baty et al. [19] to study differential equations. In order to solve the Reynolds boundary value problem, Almqvist et al. [20] studied the fundamentals of physics-informed neural networks. Neural networks with knowledge of variational physics were studied [21-22]. Scholars have studied the PINN technique in boundary layer and heat transport [23–24]. The nonlinear fluid flow problem were simulate using PINN [25-26]. Nguyen and colleagues evaluated physics-informed neural networks for non-Newtonian fluid thermo-mechanical issues [27].

Data-driven solutions of nonlinear partial differential equations utilizing PINN was studied. [28-29]. Boundary layer thermal fluid problem utilizing PINN's was presented by Hassan Bararnia et.al. [30].

The first chapter provides the foundation for understanding the major concepts and theoretical frameworks that are the pillars of the ensuing research. Basic terminology and key concepts related to the study of hybrid nanofluids, Casson fluids, and the use of machine learning techniques like Artificial Neural Networks (ANN) and Physics-Informed Neural Networks (PINN) are introduced in this chapter. In the second chapter, the MHD Casson hybrid nanofluid flow simulation across a horizontal cylinder immersed in a porous medium with slip influence is covered. This chapter introduces an Artificial Neural Network (ANN) method, where the model formulation, boundary conditions, and important parameters are discussed. The third chapter is dedicated to the use of Physics-Informed Neural Network (PINN), programmed with Python and TensorFlow, to simulate boundary layer flow and Casson fluid temperature over a horizontal cylinder with magnetic effects and porous media. The chapter is critical of the current methods and explains how PINN can be employed to address nonlinear boundary value problems in complex fluid dynamics cases. It showcases the versatility and ability of Python and TensorFlow for effectively training and solving PINN models, providing a solid solution to simulate fluid flow and temperature profiles under different physical conditions. The chapter also states the contribution of the current research, with focus on the benefits of applying PINN in simulating these intricate flow phenomena.

Table of Contents

1	Chapter 1	1
1.1	Basic Definitions	1
1.1.1	Fluid	1
1.1.2	Properties of Fluids	1
1.1.3	Types of Fluids	2
1.1.4	Classification of Fluid Flow:.....	2
1.1.5	Differential Equation of Motion	3
1.1.6	Boundary Layer Theory	4
1.1.7	Importance of Boundary Layer Theory in Fluid Dynamics	5
1.1.8	Important Dimensionless Parameters.....	6
1.2	Artificial Neural Network	7
1.2.1	Architecture of ANN.....	8
1.2.2	Types of Artificial Neural Network:	10
1.3	Physics Informed Neural Network:	12
1.3.1	General work flow of PINN:	12
	Chapter 2	15
	Artificial Neural Network Simulation of Convective Heat transfer around a Circular Cylinder.....	15
2.1	Introduction	15
2.2	Problem Formulation	16
2.2.1	Geometry of the Problem	16

2.2.2	Flow Physics.....	16
2.2.3	Transformation:.....	18
2.2.4	Physical Quantities.....	19
2.2.5	Thermo physical properties of hybrid nanofluid.....	19
2.3	Solution of Problem	20
2.3.1	First Truncation Level.....	20
2.3.2	Second Level Truncation	21
2.3.3	Third Level Truncation	22
2.4	Artificial Neural Network	24
2.5	Results and Discussion	26
2.6	Conclusion.....	55
Chapter 3		56
Physics Informed Neural Network Simulation of Non-Newtonian Flow around Cylinder		56
3.1	Introduction	56
3.2	Problem Formulation	56
3.3	Solution of the Problem	57
3.3.1	First Level Truncation:.....	57
3.3.2	Second Level Truncation	58
3.3.3	Third Level Truncation:	59
3.4	Physics Informed Neural Network	60
3.5	Results and Discussion	64

3.4 Conclusion	75
References	77

LIST OF FIGURES

Fig. 1. 1: Structure of ANN.	8
Fig. 1. 2: Block diagram of the (FNN)	10
Fig. 2.1: Physical model and coordinate system.....	17
Fig. 2.2: Workflow of ANN	25
Fig. 2.3: A visual representation of the data analysis for case 1 in scenarios 1.	40
Fig. 2.4: A visual representation of the data analysis for case 1 in scenario 2.	43
Fig. 2.5: A visual representation of the data analysis for case 1 in scenarios 3.	46
Fig. 2.6: A visual representation of the data analysis for case 1 in scenarios 4.	50
Fig. 2.7: A visual representation of the data analysis for case 1 in scenarios 5.	52
Fig. 2.8: Effects of various parameters on the Nusselt number and skin friction coefficient.	54
Fig. 3. 2 : PINN Architecture.....	61
Fig. 3. 3: PINN vs. numerical solution using 1000 epochs.....	66
Fig. 3. 4: PINN vs. numerical solution using 25000 epochs and deeper network.	66
Fig. 3. 5: Total and physics-informed loss convergence during training.	66
Fig. 3. 6: Velocity and temperature profiles for different values Pr with PINN and numerical solutions.	71
Fig. 3. 7: Velocity and temperature profiles for different values M with PINN and numerical solutions.	72
Fig. 3. 8: Velocity and temperature profiles for different values Da with PINN and numerical solutions.	73

Fig. 3. 9: Velocity and temperature profiles for different values Rd with PINN and numerical solutions.	74
---	----

LIST OF TABLES

Table 2. 1: Comparison of 1 st , 2 nd and 3 rd level truncation	23
Table 2. 2: Relevant parameters for the suggested non-Newtonian hybrid nanofluid model's flow analysis.	28
Table 2. 3: Relative assessment over backpropagation networks for scenario 1	28
Table 2. 4: Relative assessment over backpropagation networks for scenario 2 case 3.	42
Table 2. 5: Relative assessment over backpropagation networks for scenario 3	44
Table 2. 6: Relative assessment over backpropagation networks for scenario 4	48
Table 2. 7: Relative assessment over backpropagation networks for scenario 5	48
 Table 3.1: Comparison of Numerical results of 1 st , 2 nd and 3 rd level truncation	60
Table 3. 2: L_2 error between the predicted and numerical solution for different number of iterations and training steps.	67
Table 3. 3: L_2 errors of f, f', f'', θ and θ' using PINN for $\eta = 5$ and finer mesh size $\delta = 0.05$ with varying hidden layers and neurons	69
Table 3. 4: L_2 errors of f, f', f'', θ and θ' using PINN for $\eta = 5$ and finer mesh size $\delta = 0.02$ with varying hidden layers and neurons	69
Table 3. 5: L_2 errors of f, f', f'', θ and θ' using PINN for $\eta = 10$ and finer mesh size $\delta = 0.05$ with varying hidden layers and neurons	69
Table 3. 6: L_2 errors of f, f', f'', θ and θ' using PINN for $\eta = 10$ and finer mesh size $\delta = 0.02$ with varying hidden layers and neurons	69
Table 3. 7: L_2 -error of $f'(\eta)$ and $\theta(\eta)$ for varying values of Pr .	70

Table 3. 8: $L2$ -error of $f'(\eta)$ and $\theta(\eta)$ for varying values of M .	72
Table 3. 9: $L2$ -error of $f'(\eta)$ and $\theta(\eta)$ for varying values of Da .	73
Table 3. 10: $L2$ -error of $f'(\eta)$ and $\theta(\eta)$ for varying values of Rd .	74

Chapter 1

In this chapter, the basic definitions of key terms and concepts are addressed. Through this section, the readers' curiosity and interest and prepare them for a deeper dive into the subject matter in the sections that follow.

1.1 Basic Definitions

1.1.1 Fluid

Fluid is a substance that continuously deforms when subjected to a shear stress, no matter how small is that stress may be.

1.1.2 Properties of Fluids

Pressure: The symbol P stands for pressure, which is the amount of force per unit area applied in the direction normal to that area. Mathematically $P = F/A$

Temperature: The average kinetic energy of the particles in a substance is measured by a physical quantity called temperature.

Density: The mass per unit volume is the fluid's density, represented by the symbol ρ . If a mass m is contained in a volume V , then $\rho = m/V$.

Compressibility: The degree of variation in a fluid's volume under the influence of external forces is known as its compressibility. Fluids are referred to as compressible if their volume changes in response to changes in temperature or pressure; otherwise, they are referred to as incompressible fluids.

Viscosity: Viscosity of the fluid is the measure of resistance to its deformation.

Mathematically $\tau = \mu \frac{du}{dy}$, where μ is viscosity of the fluid.

1.1.3 Types of Fluids

Compressible and Incompressible fluid: A fluid is considered compressible if its density or volume varies in response to changes in pressure or temperature. If not, the fluid is referred to as incompressible.

In-viscid Fluid: An idealized fluid with zero viscosity is called an inviscid fluid because it offers no internal resistance to deformation.

Ideal Fluid: Fluid which is incompressible and in viscid is known as ideal fluid.

Real fluid: Real fluid also known as viscous fluid is a type of fluid having finite viscosity. The flow of viscous fluid is called viscous flow.

Newtonian Fluid: Newtonian fluid is a type of fluid that obeys Newton's law of viscosity ($\tau = \mu \frac{du}{dy}$). In this case shear stress is linearly related to the velocity gradient.

Non-Newtonian Fluid: Non-Newtonian fluid is a type of fluid that does not obeys Newton's law of viscosity. $\tau = K \left(\frac{du}{dy} \right)^n$, where K is the consistency index and n is flow behavior index.

1.1.4 Classification of Fluid Flow:

Compressible and Incompressible Flow: A compressible fluid flow is referred to as compressible flow, whereas an incompressible fluid flow is referred to as incompressible flow.

Ideal and Real Flow: Real or viscous flow explains the motion of fluids that exhibit viscosity and are compressible, while ideal flow depicts the motion of an ideal (inviscid and incompressible) fluid with no viscosity or thermal conductivity.

Uniform and Non-uniform Flow: If the velocity vector and other fluid characteristics stay the same throughout the flow field, the flow is considered uniform; if not, it is referred to as non-uniform flow.

Steady and Un-steady Flow: When the velocity vector and other fluid characteristics at each place in the fluid do not alter over time, the flow is considered steady. Flow is said to be un-steady when the fluid properties from point to point changes with time.

Eqn. (1.1) represent steady flow and Eqn. (1.2) represent unsteady flow.

$$\frac{\partial V}{\partial t} = \frac{\partial P}{\partial t} = \frac{\partial \rho}{\partial t} = \dots = 0 \quad (1.1)$$

$$\frac{\partial V}{\partial t} \neq \frac{\partial P}{\partial t} \neq \frac{\partial \rho}{\partial t} \neq \dots \neq 0 \quad (1.2)$$

$$\frac{\partial V}{\partial t} \neq \frac{\partial P}{\partial t} \neq \frac{\partial \rho}{\partial t} \neq \dots \neq 0 \quad (1.2)$$

Laminar Flow: A type of fluid motion known as laminar flow occurs when fluid particles travel parallel, straight, and smooth routes without coming into contact with one another. This flow produces stratified and ordered motion since the trajectories of the many particles do not overlap.

Turbulent Flow: Turbulent flow is defined as a flow in which fluid particles move erratically in all directions. The trajectories drawn by any two distinct fluid particles intersect in this kind of flow.

1.1.5 Differential Equation of Motion

Total Derivative: The total derivative $\frac{D}{Dt}$ is the rate of change of fluid property following a fluid particles. $\frac{D}{Dt} = \frac{\partial}{\partial t} + V \cdot \nabla$, where $\frac{D}{Dt}$ is the operator which act on velocity, gives the acceleration in Eulerian System and $V \cdot \nabla = u \frac{\partial}{\partial x} + v \frac{\partial}{\partial y} + w \frac{\partial}{\partial z}$

Continuity Equation: This equation state that rate of change of mass contained in the volume plus the net rate of mass flow out of the volume must be zero. Mathematically $\nabla \cdot V = -\frac{1}{\rho} \frac{D\rho}{Dt}$. In case of incompressible flow, $\frac{D\rho}{Dt} = 0$ so the continuity equation become $\nabla \cdot V = 0$.

Momentum Equation: Euler's equation is the momentum equation that results when the fluid is inviscid. The Navier-Stokes equation is the momentum equation that results when the fluid is viscous.

Euler's equation: According to this equation, the inertial forces exerted on fluid particles at any given position in an inviscid fluid's flow are equal to the total of the forces caused by gravity and pressure differences. The following is the differential form of the linear momentum equation for an in-viscid fluid in vector form:

$$\rho \frac{DV}{Dt} = -\nabla p + \rho g \quad (1.3)$$

Navier-Stokes Equation: The Navier Stokes equation describes the motion of viscous fluid. The general form of Navier Stokes equation is given as:

$$\rho \frac{DV}{Dt} = -\nabla p + \rho g + \mu \nabla^2 V. \quad (1.4)$$

❖ **Energy Equation:** The first law of thermodynamics generally known as the law of conservation of energy, mathematically form is

$$\rho C_p \left(\frac{\partial T^*}{\partial t} + V^* \cdot \nabla T^* \right) = k \nabla^2 T^* + \mu \phi \quad (1.5)$$

1.1.6 Boundary Layer Theory

The boundary layer is defined as the thin layer of the flow on the boundary within which the velocities changing from zero at the solid boundary to the free stream velocity in the direction normal to the boundary. In the boundary layer the velocity gradient is large

and the shear exerted by the fluid is given as $\tau = \mu \frac{du}{dy}$. Outside the boundary, velocity is constant and velocity gradient is zero and hence shear stress is zero. Imagine a fluid flowing laminarly in two dimensions across a fixed semi-infinite flat plate with a high Reynolds number and low viscosity. In contrast to ideal fluid flow, which would cause the fluid to slide across the surface, a real fluid clings to the plate and exhibits the no-slip condition. The fluid velocity at the surface is likewise 0 because the plate is motionless. In the direction corresponding to the surface, the fluid velocity progressively rises as one advances away from the plate. When the velocity is far enough away from the plate, it asymptotically gets closer to the free-stream velocity U . Known as the boundary layer, this narrow area close to the plate is where the velocity shifts from zero to U .

1.1.7 Importance of Boundary Layer Theory in Fluid Dynamics

The boundary layer flow is significant in fluid dynamics because it controls the interaction between a solid surface and the surrounding fluid, influencing drag, heat transfer, and mass transport significantly. It helps in the understanding of skin friction drag, which directly influences the efficiency of vehicles and aircraft. The phenomena is also crucial in heat transfer analysis, especially thermal boundary layers, which effect cooling and heating operations in engineering systems. Also, separation of the boundary layer, where flow separates from the surface, creates greater pressure drag and flow instability and negatively affects turbine, aircraft wing, and submarine performances. Some basic definition are:

Boundary Layer Thickness: The distance between a solid boundary, like a surface, and the point in a fluid flow when the fluid velocity approaches ninety percent of the free stream velocity is referred to as the boundary layer thickness. It denotes the area of

the fluid in which the viscosity impacts are most prominent resulting in a gradient in velocity from the boundary (fluid sticks to the wall) to the boundary layer's outer edge. Grasp and evaluating fluid dynamics near surfaces—which affect heat transfer, drag, and overall flow characteristics—needs a grasp of the term of boundary layer thickness.

Displacement Thickness: Displacement thickness, denoted as δ is a measure of the reduction in flow rate due to the presence of the boundary layer. It represents the distance by which the external inviscid flow is displaced outward due to the slowing down of the fluid near the boundary.

$$\delta = \int_0^{\infty} (U_e - u) dy \quad (1.6)$$

Momentum Thickness: The loss of momentum in the boundary layer compared to the free-stream flow.

$$\theta = \int_0^{\infty} \frac{U_e}{u} (U_e - u) dy \quad (1.7)$$

1.1.8 Important Dimensionless Parameters

Reynold number: The Reynold number (Re) is a dimensionless quantity that helps predict fluid flow patterns in different in different situation by measuring the ratio between inertial and viscous forces. At low Reynold numbers, flow tends to be dominated by laminar, while at high Reynolds number, flow tends to be turbulent.

$$Re = \frac{\rho V^* d^*}{\mu} = \frac{V^* d^*}{\nu} \quad (1.8)$$

Prandtl Number: The Prandtl number (Pr) is a dimensionless number that represents the ratio of momentum diffusivity to thermal diffusivity in fluid. Mathematically Prandtl number expressed as:

$$Pr = \frac{\nu}{\alpha} = \frac{\mu/\rho}{k/\rho c_p} = \frac{\mu c_p}{k} \quad (1.9)$$

Nusselt Number: The ratio of convective to conductive heat transport in a fluid is represented by the dimensionless Nusselt number (Nu). The definition of the Nusselt number in mathematics is:

$$Nu = \frac{h^* L^*}{k^*} \quad (1.10)$$

Grashof number: The Grashof number (Gr) is a dimensionless number that which approximates the ratio of the buoyancy to viscous forces acting on a fluid. Mathematically expressed as;

$$Gr = \frac{gL^3 \beta \Delta T}{\nu^2} \quad (1.11)$$

Casson fluid model: The Casson fluid model is a non-Newtonian fluid model used to describe the properties of fluids that contain yield stress. Unlike Newtonian fluids, where there is a linear relationship between shear stress and shear rate, Casson fluids must have a certain minimum shear stress (yield stress, τ_y) before it begins to flow. Once this threshold is reached, the fluid behaves in a shear-thinning manner, i.e., its viscosity decreases with an increasing shear stress. Casson fluid are used in blood flow modeling, food processing, polymer solution etc. The Mathematical model of Casson fluid are described as:

$$\tau^{\frac{1}{2}} = \tau_y^{\frac{1}{2}} + \eta \dot{\gamma}^{\frac{1}{2}} \quad (1.12)$$

τ the shear stress, τ_y is the yield stress (minimum stress required for fluid flow), η is the plastic viscosity and $\dot{\gamma}$ is the shear rate.

1.2 Artificial Neural Network

ANN form the basic building block of contemporary artificial intelligence and machine learning. Based on the architecture of biological neural networks in the human brain,

ANNs are made up of interlinked nodes (neurons) laid out in layers that handle data by mimicking the functioning of biological neurons passing signals. While abridged compared to the human brain, ANNs can learn from data, identify patterns, and predict outcomes. The theoretical basis for ANNs was originally presented in 1943, being an important step towards the creation of intelligent computational models.

1.2.1 Architecture of ANN

Neural activity in the human brain can be simulated to create artificially intelligent systems. The information processing architecture of this network is its primary characteristic. ANNs function through information processing, a method similar to that of the human brain. Several networked neurons, or processing units, make up this system, which works together to do specific tasks at the same time. Neural computing is a paradigm for mathematics that draws inspiration from biological principles. The neural architecture of the human brain is replicated by this computer model.

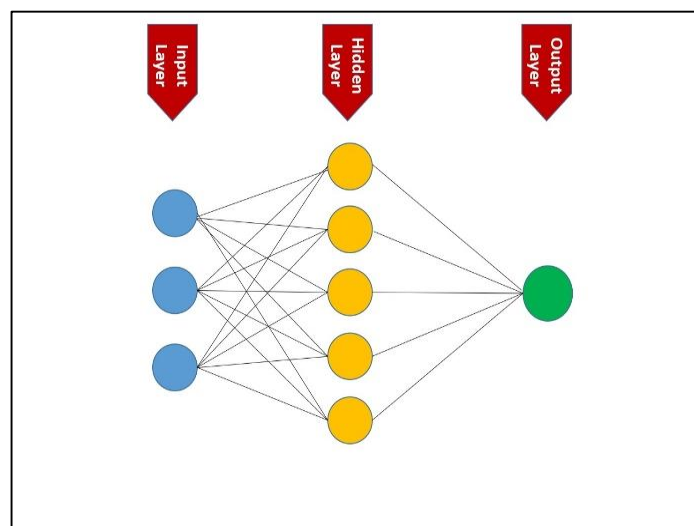


Fig. 1. 1: Structure of ANN.

It is made up of layers of interconnected nodes, or neurons. Based on the data these nodes process, the network adjusts their weights (connection strengths) throughout

training. Consequently, the network can execute a range of machine learning and artificial intelligence tasks, including pattern identification and outcome prediction. Following the connection of the hidden layers, the "output layer," where the result is output, is displayed in Fig. 1.1.

Neurons (Nodes): Simple processing units called neurons (also known as nodes) take in input data, apply weight, add bias, and then run the outcome via an activation function.

Input Layer: Raw input data is sent to the next layers by the first layer. It does not do any calculations; it merely distributes the inputs.

Hidden Layers: The intermediate levels that lie between the input and output layers are known as hidden layers. They perform intricate calculations to generate more abstract representations of the input data using the weights, biases, and activation functions. The complexity of the issue will determine how many hidden layers and neurons are used.

Output Layer: The output layer is the final layer that produces the network's output.

Weights: Weights are the parameters for adjusting the input signals, the weight given to each neuronal connection determines the importance of the input.

Bias: This parameter is applied prior to the activation function being applied to the weighted sum of the inputs, allowing the activation function to be shifted to the left or right, increasing the model's adaptability.

Activation Function: An activation function, also known as a transfer function, applies a transformation to the input of a neuron to determine its output. Usually, this transformation squashes the output to a range like 0 to 1 or -1 to 1. Common varieties

include Gaussian, piecewise linear, unit step, sigmoid (unipolar and bipolar), and hyperbolic tangent functions.

1.2.2 Types of Artificial Neural Network:

An ANN's architecture greatly influences its capacity to recognize intricate patterns, generalize to previously unobserved data, and achieve high predicted accuracy. There are several types of ANNs, such as feedforward neural networks, recurrent neural networks, convolutional neural networks, and more, each of which is appropriate for a particular set of data and tasks.

Feed-Forward Neural Network: The feedforward neural network (FNN) is a basic kind of artificial neural network (ANN) in which the connections between the neurons do not cycle. Information flows from the input layer to the output layer, via any hidden layers, and then to the output layer. Network connections to the same or earlier tiers are prohibited in an FFN, where data flows strictly feed-forward from the input node to the output node. The feedforward neural network (FNN) block diagram is shown in Fig. 1.2.

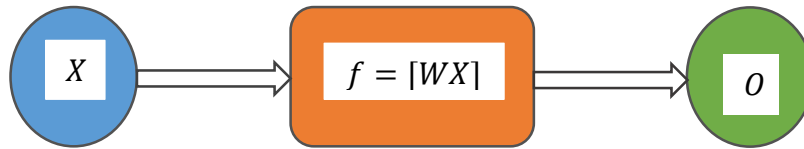


Fig. 1. 2: Block diagram of the (FNN)

Where $X = (x_1, x_2, \dots, x_n)$ denotes the input vector, $O = (o_1, o_2, \dots, o_m)$ denotes the output vector, and f is the activation function. The symbol $W = w_{ij}$, represents

the weight matrix or connection matrix, and the scalar product of the input vectors and the weight vectors, w_{ij} . is the net input value, or WX .

Feedback Neural Networks: Feedback neural networks, sometimes referred to as recurrent neural networks (RNNs), are artificial neural networks (ANNs) where feedback loops are made possible by the cycles created by connections between neurons. By using loops, signals in this network can move in both directions. Although this network is very strong, it can occasionally become very confusing. Neuronal connections of any kind are allowed. By maintaining internal state memory, recurrent neural networks (RNNs) are able to exhibit dynamic temporal behavior in contrast to feedforward neural networks, which only transport information from input to output. This neural network starts with the same front propagation as a feed-forward network and saves all of the data it has processed for subsequent use. Backpropagation lets the system self-learn and keep trying until it gets the right forecast if the network's initial prediction turns out to be incorrect. Feedback neural networks are used to find the optimal configuration of interdependent variables in optimization problems. Because they are dynamic, their states change continuously until they reach equilibrium

Backpropagation Neural Network (BPNN): A Backpropagation Neural Network (BPNN) is an artificial neural network that uses the backpropagation technique for training. It is composed of an input layer, an output layer, and one or more hidden layers. The forward pass calculates the network's output, while the backward pass modifies the weights and propagates the error gradient back through the network to minimize the loss function. BPNNs are widely used for applications such as pattern recognition, regression, and classification because of their ability to identify complex patterns in data.

1.3 Physics Informed Neural Network

Physics-Informed Neural Networks (PINN) are a type of scientific machine learning models that explicitly integrate physical laws, typically represented as partial differential equations (PDEs) or ordinary differential equations (ODEs) into artificial neural network training. This method has gained a lot of traction in many branches of computational science and engineering, particularly in fields where labeled data or high processing costs limit the use of classic numerical methods. While traditional neural networks mainly rely on supervised learning with large datasets, the governing equations of many physical systems, particularly fluid mechanics, heat transfer, and boundary layer theory, are often well understood, and PINN take advantage of this prior knowledge by embedding the physics into the neural network's loss function, which enables the network to learn solutions that naturally satisfy the underlying physical laws, even in the absence of extensive training data.

PINN are especially helpful in tackling inverse problems (inferring unknown parameters or inputs from observed outputs), forward problems (predicting the state of a system given initial/boundary conditions and parameters), and data assimilation tasks within a single, cohesive framework.

1.3.1 General work flow of PINN:

The implementation of a PINN for solving a physical problem involves several key components. The general workflow is outlined below:

Problem Formulation:

- i. Determine the governing equations: These may be ODEs or PDEs formulated from conservation laws, constitutive relations, or empirical

models. For example, in boundary layer theory, it is common to work with nonlinear ODEs that govern the momentum and energy transport.

- ii. Specify the solution domain: This is the spatial and/or time domain over which the solution should be approximated.
- iii. Identify boundary and initial conditions: These are essential for a properly posed problem and are imposed within the PINN framework.

Neural Network Design: A fully connected feed-forward neural network (FNN) is constructed:

- i. Input layer: The input layer receives the data or coordinates in the form of spatial location x, η or time t .
- ii. Hidden layer: The hidden layers consist of neurons with nonlinear activation function such as tanh, sigmoid, ReLU, which enable the network to approximate complex, non-linear mappings.
- iii. Output layer: The output layer provides the predicted values of the solution variables, such as velocity $f(\eta_i)$, temperature $\theta(\eta_i)$, or concentration.

Collocation and Boundary Points: To train the model, a set of collocation points within the domain is selected. These are the points where the governing equations are enforced. Additionally, boundary points are sampled to enforce the prescribed conditions. Unlike traditional numerical methods, PINNs do not require grid-based discretization and are thus mesh-free.

Loss Function Formulation: The total loss function contains several components:

- i. **Physics loss:** Quantifies the residual of the governing equations at the collocation points. This is computed using automatic differentiation, which allows for exact gradients of the network output with respect to inputs.
- ii. **Boundary condition loss:** Measures the discrepancy between the network prediction and the known boundary or initial values.
- iii. **Data loss (if applicable):** Incorporates any available empirical data to further guide the learning process.

The total loss function will be:

$$L_{Total} = L_{Physics} + L_{BC} + L_{Data} \quad (1.13)$$

Optimization and Training: The network weights and biases are adjusted by minimizing the entire loss function through gradient-based optimization methods like Adam for initial convergence and L-BFGS for precise, high-accuracy training. The training is iterative and goes on until the loss function converges to a low value, which means that the physical constraints and boundary conditions are being met.

Prediction and Post-Processing: After training, the PINN can be employed to predict at any location within the domain. The solution is continuous and differentiable, which is beneficial for subsequent analysis, for instance, calculating derived quantities. PINNs also allow for parameter inference and model validation when inverse problems are formulated.

Chapter 2

Artificial Neural Network Simulation of Convective Heat transfer of hybrid nanofluid flow around a Circular Cylinder

2.1 Introduction

The current chapter investigates the magnetohydrodynamic (MHD) boundary layer flow of Casson hybrid nanofluids (CHNFs) over a porous cylinder with thermal radiation and velocity slip using an Artificial Neural Network approach. Al_2O_3 and CuO hybrid nanoparticles floating in a Casson base fluid form the basis of the Tiwari-Das nanofluid model, which is used to develop the flow equations. Reference [31] is consulted for the appropriate physical correlations for viscosity, electrical conductivity, thermal conductivity, and specific heat capacity. Suitable non-similar transformations are used to convert dimensional equations into a dimensionless form. A set of ordinary differential equations is obtained by applying the Local Non-Similarity (LNS) technique up to the third truncation level in order to simplify the system. To obtain the reference solution, the system of Eqns. are then solved by MATLAB's built-in solver, bvp4c. Supervised machine learning methods based on Artificial Neural Network (ANN) simulations are used to calculate an approximate solution. A comparison of the reference solution and the ANN's anticipated solution reveals that they are well suited. The effect of dimensionless parameter M (magnetic parameter), Da (Darcy number), β (Casson parameter) S_f (velocity slip parameter) and S_T (thermal jump parameter) on momentum boundary layer flow are examined.

2.2 Problem Formulation

2.2.1 Geometry of the Problem

The flow geometry of the problem under consideration is shown in Fig. 2.1. In the cylinder, the x coordinate is displayed in its tangential direction, whereas the y coordinate is normal to the surface. A horizontal cylinder's radius is represented by the letter a , which is parallel to the y -axis. The angle of y -axis with regard to the vertical ($0 \leq \phi \leq \pi$) is given by $\phi = \frac{x}{a}$.

2.2.2 Flow Physics

Consider the steady, two dimensional MHD flow of an incompressible, electrically conducting Casson hybrid nanofluid over a horizontal permeable circular cylinder with saturated porous medium and thermal radiation. The magnetic field is uniform having magnitude B_0 and is applied in radial direction. The gravitational force g is acting in downward direction. It is assume that Boussinesq approximation holds. Let T_w is constant temperature and T_∞ is ambient temperature of the fluid. The governing boundary layer equation are follow as [2]

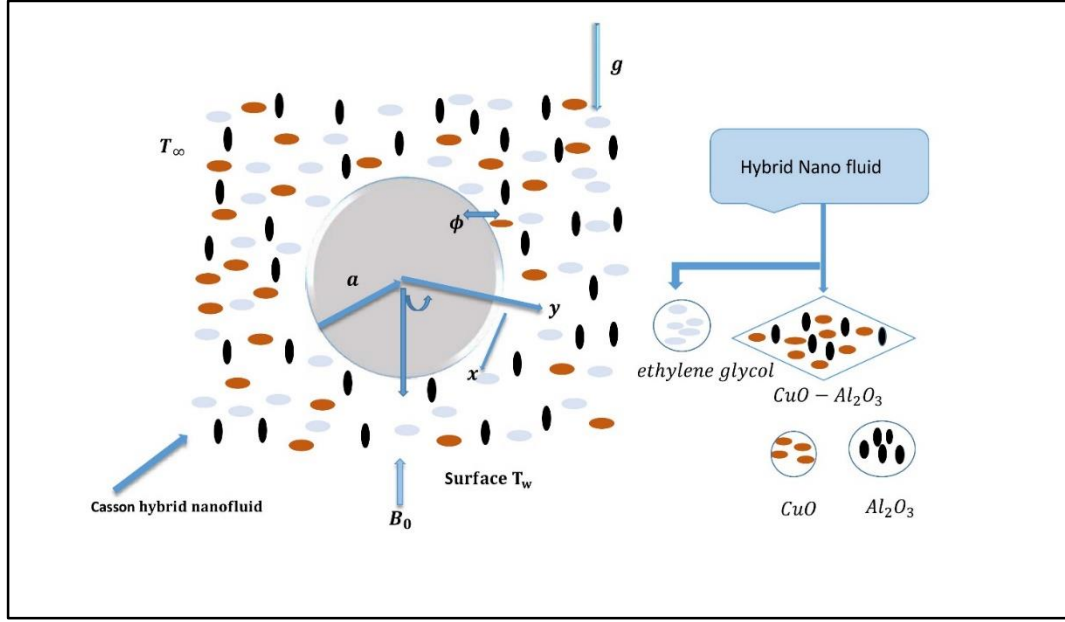


Fig. 2.1: Physical model and coordinate system

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0, \quad (2.1)$$

$$u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = \nu_{hnf} \left(1 + \frac{1}{\beta} \right) \frac{\partial^2 u}{\partial y^2} - \frac{\sigma_{hnf} B_0^2}{\rho_{hnf}} u - \frac{\nu_{hnf}}{K} u - cu^2 + g \frac{(\rho \beta_1)_{hnf}}{\rho_{hnf}} (T - T_\infty) \sin\left(\frac{x}{a}\right), \quad (2.2)$$

$$u \frac{\partial T}{\partial x} + v \frac{\partial T}{\partial y} = \alpha_{hnf} \frac{\partial^2 T}{\partial y^2} - \frac{1}{(\rho c_p)_{hnf}} \frac{\partial q_x}{\partial y}. \quad (2.3)$$

Whereas thermal radiation is denoted by

$$\frac{\partial q_x}{\partial y} = -\frac{16\sigma^* T_\infty^3}{3k^*} \left(\frac{\partial^2 T}{\partial y^2} \right).$$

In Eqns. (2.2) and (2.3), u and v represent the velocity components in the x and y directions, respectively, β is the non-Newtonian Casson parameter and ν is the conducting fluid's kinematic viscosity. α Stands for thermal diffusivity, T for temperature, and K and c for the porous medium's permeability and inertia coefficient, respectively. T_∞ - the free stream temperature, and β_f is the coefficient of thermal expansion.

The slip boundary condition are as follow [31].

$$\text{At } y = 0, u = N_0 \left(1 + \frac{1}{\beta}\right) \frac{\partial u}{\partial y}, v = 0, T = T_w + k_0 \frac{\partial T}{\partial y}, \quad (2.4)$$

$$\text{As } y \rightarrow \infty, u \rightarrow 0, T \rightarrow T_\infty.$$

Where N_0 the velocity is slip factor and K_0 is the thermal slip factor. For $N_0=0=K_0$, one can recover the no slip case.

2.2.3 Transformation

Similarity transformation is a powerful mathematical techniques employed to reduce the complexity of partial differential equations (PDEs), typically arising in fluid mechanics problems. The reduction is achieved by introducing dimensionless variables.

In current study the dimensionless variables is used from [2] as follow:

$$\begin{aligned} \xi &= \frac{x}{a}, \quad \eta = \frac{y}{a} \sqrt[4]{Gr}, f(\xi, \eta) = \frac{\psi}{v_f \xi \sqrt[4]{Gr}}, u = \frac{\partial \psi}{\partial y}, v = -\frac{\partial \psi}{\partial x}, \theta(\xi, \eta) \\ &= \frac{T - T_\infty}{T_w - T_\infty}, \\ Gr &= \frac{g \beta_f (T_w - T_\infty) a^3}{v_f^2}. \end{aligned} \quad (2.5)$$

In view of dimensionless variables Eqns. (2.5), Eqns. (2.2)-(2.4) reduce to the following coupled, nonlinear, dimensionless partial differential equations:

$$\left(\frac{v_{hnf}}{v_f}\right) \left(1 + \frac{1}{\beta}\right) f''' + f f'' - (1 + \xi \Lambda) f'^2 - \left(\left(\frac{\sigma_{hnf}}{\sigma_f}\right) \left(\frac{\rho_f}{\rho_{hnf}}\right) M + \right. \quad (2.6)$$

$$\left.\left(\frac{v_{hnf}}{v_f}\right) \frac{1}{Da}\right) f' + \frac{\sin \xi}{\xi} \left[\left(\frac{(\rho \beta)_{hnf}}{(\rho c_p)_{hnf}}\right) \theta \right] = \xi \left(f' \frac{\partial f'}{\partial \xi} - f'' \frac{\partial f}{\partial \xi} \right),$$

$$\frac{1}{pr} \left[\left(\frac{\alpha_{hnf}}{\alpha_f}\right) + \left(\frac{(\rho c_p)_f}{(\rho c_p)_{hnf}}\right) \frac{4}{3} Rd \right] \theta'' + f \theta' = \xi \left(f' \frac{\partial \theta}{\partial \xi} - \frac{\theta' \partial f}{\partial \xi} \right). \quad (2.7)$$

Dimensionless boundary conditions are as follow

$$\text{at } \eta = 0, f' = S_f \left(1 + \frac{1}{\beta}\right) f''(0), f = 0, \theta = 1 + S_T \theta'(0), \quad (2.8)$$

$$\text{As } \eta \rightarrow \infty, f' \rightarrow 0, \theta \rightarrow 0.$$

The physical parameter that arises in Eqns. (2.6)-(2.8) are given below

$$M = \frac{\sigma_f \beta_0^2 a^3}{\rho_f \nu_f \sqrt{Gr}}, Pr = \frac{\nu_f}{\alpha_f}, Da = \frac{\kappa \sqrt{Gr}}{a^2}, \Lambda = Ca, Rd = \frac{4\sigma^* T_\infty^3}{k^* k_f},$$

$$S_T = \frac{\kappa_0 \sqrt[4]{Gr}}{a}, S_f = \frac{N_0 \sqrt[4]{Gr}}{a}.$$

In above equations, the prime denote the differentiation with respect to η , η is the dimensionless radial coordinates, and ξ is the dimensionless tangential coordinate, Λ is the local inertia coefficient (Forchheimer parameter), Da is the Darcy parameter, M is the magnetic parameter and Gr is the Grashof (free convection) parameter. Pr is the Prandtl number, Rd is the radiation parameter, S_f and S_T are the dimensional velocity and thermal slip parameter respectively.

2.2.4 Physical Quantities

Physical quantities i.e. Skin friction and Nusselt number are presented below

$$\frac{1}{2} C_f \sqrt[4]{Gr} = \left(1 + \frac{1}{\beta}\right) \xi f''(\xi, 0), \quad (2.9)$$

$$Nu \sqrt[4]{Gr} = -\theta'(\xi, 0). \quad (2.10)$$

2.2.5 Thermo physical properties of hybrid nanofluid

Thermo physical properties of hybrid nanofluid are follow as [32];

$$\mu_{hnf} = \frac{\mu_f}{[(1-\varphi_1)(1-\varphi_2)]^{2.5}},$$

$$\rho_{hnf} = (1 - \varphi_2)\{(1 - \varphi_1)\rho_f + \varphi_1\rho_1\} + \varphi_2\rho_2,$$

$$(\rho c_p)_{hnf} = (1 - \varphi_2)\{(1 - \varphi_1)(\rho c_p)_f\} + \varphi_2(\rho c_p)_2,$$

$$(\rho\beta)_{hnf} = (1 - \varphi_2)\{(1 - \varphi_1)(\rho\beta)_f + \varphi_1(\rho\beta)_1\} + \varphi_2(\rho\beta)_2,$$

$$\frac{\sigma_{hnf}}{\sigma_f} = \left[1 + \frac{3\left(\frac{\sigma_2}{\sigma_f} - 1\right)\varphi_2}{\left(\frac{\sigma_2}{\sigma_f} + 2\right) - \left(\frac{\sigma_2}{\sigma_f} - 1\right)\varphi_2} \right], \quad \frac{\sigma_{nf}}{\sigma_f} = \left[1 + \frac{3\left(\frac{\sigma_1}{\sigma_f} - 1\right)\varphi_2}{\left(\frac{\sigma_1}{\sigma_f} + 2\right) - \left(\frac{\sigma_1}{\sigma_f} - 1\right)\varphi_2} \right],$$

$$\frac{\alpha_{hnf}}{\alpha_f} = \frac{2\alpha_{nf} - (\alpha_{nf} - \alpha_2)\varphi_2 + \alpha_2}{(\alpha_{nf} - \alpha_2)\varphi_2 + 2\alpha_{nf} + \alpha_2}, \quad \frac{\alpha_{nf}}{\alpha_f} = \frac{2\alpha_f - (\alpha_{nf} - \alpha_1)\varphi_1 + \alpha_1}{(\alpha_f - \alpha_1)\varphi_1 + 2\alpha_f + \alpha_1}.$$

2.3 Solution of Problem

The system of non-linear PDEs are solve using the Local Non-Similarity solution method. The local non-similarity solution was developed by Sparrow and coworker and has been since applied by many researches to solve various non-similar boundary layer problem. To employ the local non similarity techniques to the problem under consideration, we follow [33] and [34] as:

2.3.1 First Truncation Level

The term on right hand side of Eqns. (2.6)-(2.7) are neglected under the first level truncation i.e. $\xi \ll 1$. Consequently the terms involving $\xi \frac{\partial(\cdot)}{\partial \xi}$ are small. The system of Eqns. (2.6)-(2.7) subject to this truncation get the following forms:

$$\left(\frac{v_{hnf}}{v_f}\right)\left(1 + \frac{1}{\beta}\right)f''' + ff'' - (1 + \xi\Lambda)f'^2 - \left(\left(\frac{\sigma_{hnf}}{\sigma_f}\right)\left(\frac{\rho_f}{\rho_{hnf}}\right)M + \right. \quad (2.11)$$

$$\left.\left(\frac{v_{hf}}{v_f}\right)\frac{1}{Da}\right)f' + \frac{\sin\xi}{\xi}\left[\left(\frac{(\rho\beta)_{hnf}}{(\rho c_p)_{hnf}}\right)\theta\right] = 0,$$

$$\frac{1}{pr}\left[\left(\frac{\alpha_{hnf}}{\alpha_f}\right) + \left(\frac{(\rho c_p)_f}{(\rho c_p)_{hnf}}\right)\frac{4}{3}Rd\right]\theta'' + f\theta' = 0, \quad (2.12)$$

$$\begin{aligned} \text{at } \eta = 0, f' = S_f \left(1 + \frac{1}{\beta}\right) f''(0), f = 0, \theta = 1 + S_T \theta'(0), \\ \text{As } \eta \rightarrow \infty, f' \rightarrow 0, \theta \rightarrow 0. \end{aligned} \quad (2.13)$$

2.3.2 Second Level Truncation

To derive equation of higher order truncation, the following functions are defined as:

$$g = \frac{\partial f}{\partial \xi}, \quad h = \frac{\partial g}{\partial \xi}, \quad \phi = \frac{\partial \theta}{\partial \xi}, \quad \chi = \frac{\partial \phi}{\partial \xi} \quad (2.14)$$

Secondary equation for g and ϕ and their boundary condition are obtained by taking derivative of Eqns. (2.6)-(2.7) and boundary condition Eqn. (2.8) with respect to ξ .

$$\left(\frac{v_{hnf}}{v_f}\right) \left(1 + \frac{1}{\beta}\right) f''' + f f'' - (1 + \xi \Lambda) f'^2 - \left(\left(\frac{\sigma_{hnf}}{\sigma_f}\right) \left(\frac{\rho_f}{\rho_{hnf}}\right) M + \right. \quad (2.15)$$

$$\left.\left(\frac{v_{hnf}}{v_f}\right) \frac{1}{Da}\right) f' + \frac{\sin \xi}{\xi} \left[\left(\frac{(\rho \beta)_{hnf}}{(\rho c_p)_{hnf}}\right) \theta \right] = \xi (f' g' - f'' g),$$

$$\frac{1}{pr} \left[\left(\frac{\alpha_{hnf}}{\alpha_f}\right) + \left(\frac{(\rho c_p)_f}{(\rho c_p)_{hnf}}\right) \frac{4}{3} Rd \right] \theta'' + f \theta' = \xi (f' \theta - \theta' g), \quad (2.16)$$

$$\left(\frac{v_{hnf}}{v_f}\right) \left(1 + \frac{1}{\beta}\right) g''' + f g'' + 2 f'' g - f' g' - (1 + \xi \Lambda) g'^2 - \Lambda f' - \quad (2.17)$$

$$\left(\left(\frac{\sigma_{hnf}}{\sigma_f}\right) \left(\frac{\rho_f}{\rho_{hnf}}\right) M \left(\frac{v_{hnf}}{v_f}\right) \frac{1}{Da}\right) g' + \left(\frac{(\rho \beta)_{hnf}}{(\rho c_p)_{hnf}}\right) \left[\left(\frac{\sin \xi}{\xi} \phi\right) \left(\frac{\xi \cos(\xi) - \sin(\xi)}{\xi^2}\right) \theta \right] =$$

$$\xi (g' g' - g'' g),$$

$$\frac{1}{pr} \left[\left(\frac{\alpha_{hnf}}{\alpha_f}\right) + \left(\frac{(\rho c_p)_f}{(\rho c_p)_{hnf}}\right) \frac{4}{3} Rd \right] \phi'' + f \phi' - f' \phi + 2 \theta' g = \xi (g' \phi - \phi' g), \quad (2.18)$$

Boundary conditions for second level of truncation are describe as;

$$\begin{aligned} \text{at } \eta = 0, \quad f' = S_f \left(1 + \frac{1}{\beta}\right) f''(0), \quad f = 0, \quad \theta = 1 + S_T \theta'(0), \quad g' = S_f \left(1 + \frac{1}{\beta}\right) g''(0), \end{aligned} \quad (2.19)$$

$$g = 0, \phi = S_T \phi'(0),$$

$$\text{As } \eta \rightarrow \infty, f' \rightarrow 0, \theta \rightarrow 0, g' \rightarrow 0, \phi \rightarrow 0.$$

2.3.3 Third Level Truncation

The conservation equations for f and θ function and their corresponding equations for g and ϕ are preserved without approximation at the third level of truncation. The process of determining the first derivative of the equations (2.15) - (2.16) generates additional subsidiary equations for h and χ as well as boundary conditions. $\xi \ll 1$ is approximated to arrive at the system of equations for the third degree of truncation. So the terms involving $\xi \frac{\partial h}{\partial \xi}$, $\xi \frac{\partial h'}{\partial \xi}$ and $\xi \frac{\partial \chi}{\partial \xi}$ are small. The system of Eqns. Subject to this truncation get the following form:

$$\left(\frac{v_{hnf}}{v_f}\right) \left(1 + \frac{1}{\beta}\right) f''' + f f'' - (1 + \xi \Lambda) f'^2 - \left(\left(\frac{\sigma_{hnf}}{\sigma_f}\right) \left(\frac{\rho_f}{\rho_{hnf}}\right) M + \right. \quad (2.20)$$

$$\left.\left(\frac{v_{hnf}}{v_f}\right) \frac{1}{Da}\right) f' + \frac{\sin \xi}{\xi} \left[\left(\frac{(\rho \beta)_{hnf}}{(\rho c_p)_{hnf}}\right) \theta \right] = \xi (f' g' - f'' g),$$

$$\frac{1}{pr} \left[\left(\frac{\alpha_{hnf}}{\alpha_f}\right) + \left(\frac{(\rho c_p)_f}{(\rho c_p)_{hnf}}\right) \frac{4}{3} Rd \right] \theta'' + f \theta' = \xi (f' \theta - \theta' g), \quad (2.21)$$

$$\left(\frac{v_{hnf}}{v_f}\right) \left(1 + \frac{1}{\beta}\right) g''' + f g'' + 2 f'' g - f' g' - (1 + \xi \Lambda) g'^2 - \Lambda f' - \quad (2.22)$$

$$\left(\left(\frac{\sigma_{hnf}}{\sigma_f}\right) \left(\frac{\rho_f}{\rho_{hnf}}\right) M + \left(\frac{v_{hnf}}{v_f}\right) \frac{1}{Da}\right) g' +$$

$$\left(\frac{(\rho \beta)_{hnf}}{(\rho c_p)_{hnf}}\right) \left[\left(\frac{\sin \xi}{\xi}\right) \phi \right] \left(\frac{\xi \cos(\xi) - \sin(\xi)}{\xi^2}\right) \theta \right] = \xi (g' g' - g'' g + f' h' - f'' h),$$

$$\frac{1}{pr} \left[\left(\frac{\alpha_{hnf}}{\alpha_f}\right) + \left(\frac{(\rho c_p)_f}{(\rho c_p)_{hnf}}\right) \frac{4}{3} Rd \right] \phi'' + f \phi' - f' \phi + 2 \theta' g = \xi (g' \phi - \phi' g + \quad (2.23)$$

$$f' \chi - \theta' h),$$

$$\left(\frac{v_{hnf}}{v_f}\right)\left(1 + \frac{1}{\beta}\right)h''' + fh'' + 3f''h + 4g''g - 2f'h' - 2g'g' - (1 + \quad (2.24)$$

$$\xi\Lambda)h'^2 - 2\Lambda g'^2 - \left(\left(\frac{\sigma_{hnf}}{\sigma_f}\right)\left(\frac{\rho_f}{\rho_{hnf}}\right)M + \left(\frac{v_{hnf}}{v_f}\right)\frac{1}{Da}\right)h' +$$

$$\left(\frac{(\rho\beta)_{hnf}}{(\rho c_p)_{hnf}}\right)\left[2\left(\frac{\xi \cos(\xi) - \sin(\xi)}{\xi^2}\right)\phi + \frac{\sin\xi}{\xi}\chi - \left(\frac{\xi^2 \sin\xi + 2\xi \cos\xi - 2\sin\xi}{\xi^3}\right)\theta =$$

$$\xi(3h'g' - h''g - 2g''h),$$

$$\frac{1}{pr}\left[\left(\frac{\alpha_{hnf}}{\alpha_f}\right) + \left(\frac{(\rho c_p)_f}{(\rho c_p)_{hnf}}\right)\frac{4}{3}Rd\right]\chi'' - 2\phi'g + f\chi' - 2g'\phi - 2f'\chi + \theta'h = \quad (2.25)$$

$$\xi(2g'\chi - \chi'g + h'\phi - 2\phi'h).$$

Boundary condition for third level of truncation are given below;

$$at \eta = 0, \quad f' = S_f(1 + \frac{1}{\beta})f''(0), \quad f = 0, \quad \theta = 1 + S_T\theta'(0), \quad g' = S_f(1 + \frac{1}{\beta})g''(0), \quad (2.26)$$

$$g = 0, \phi = S_T\phi'(0) \quad h = 0, h' = S_f(1 + \frac{1}{\beta})h''(0), \chi = S_T\chi'(0),$$

$$As \eta \rightarrow \infty, f' \rightarrow 0, \theta \rightarrow 0, g' \rightarrow 0, \phi \rightarrow 0, h' \rightarrow 0, \chi \rightarrow 0.$$

Table 2. 1: Comparison of 1st, 2nd and 3rd level truncation

		1 st Level Truncation		2 nd Level Truncation		3 rd Level Truncation	
M	η	$f'(\eta)$	$\theta(\eta)$	$f'(\eta)$	$\theta(\eta)$	$f'(\eta)$	$\theta(\eta)$
0.6	0	0.039457	1.222876	0.037808	0.708338	0.032126	0.705263
0.6	1	0.28513	0.940597	0.201848	0.384346	0.161920	0.382956
0.6	2	0.329241	0.689143	0.164008	0.185686	0.124797	0.176915
0.6	3	0.28705	0.46254	0.093917	0.070168	0.071069	0.073208
0.6	4	0.218836	0.296122	0.046245	0.022563	0.034546	0.027471
0.6	5	0.146061	0.170395	0.01907	0.005172	0.015471	0.009913
0.6	6	0.08415	0.086437	0.006894	0.000570	0.006354	0.003328

0.6	7	0.037761	0.034777	0.002149	0.000165	0.002118	0.000877
0.6	8	0	0	0	0	0	0

2.4 Artificial Neural Network

An estimated solution to the current problem is investigated using an ANN approach, a machine learning algorithm. The Multilayer Perceptual (MLP) method is a well-known illustration of an ANN algorithm. It is essential to use the MLP-ANN structure in order to get a more precise response. Fig. 2 displays the general flowchart utilized in the design for (MLP-ANN) schemes. Three tiers made up the MLP-ANN system. The input data is sent to level 1, also known as the input layer. The hidden layer, the second level, examines the neurons that comprise the calculation. There could be one or more hidden layers in an MLP system. The output layer, sometimes referred to as the third or output layer, shows the outcomes of the forecasts. To reduce the difference between the target's estimation and the actual estimation, the data that is sent upstream from the input layer is sent back into the input layer via back propagation. For as long as the error rate is kept to a minimum, this process continues. This process ends when the MLP-ANN training is finished and the minimum prediction productivity is attained. The dataset is separated into testing, training, and validation phases in order to use the MLP-ANN method. Seventy percent of the data is used for training, with the remaining twenty percent being used for results validation and testing. In this study, we employed ANN to predict the output in the output layer using tan Sig and Purelin [35] as the activation function in a hidden layer. To improve the model's accuracy, we employed 36 neurons and 10 hidden layers in the current investigation. The transport function can be represented mathematically as follows:

$$f(x) = \frac{1}{1+e^{-x}}, \quad \text{Purelin}(x) = x, \quad (2.27)$$

The next phase of the MLP-ANN system's development will evaluate the forecasting model's accuracy. Mean square error (MSE), average relative error (ARE), and correlation coefficient (R) are the variables to be studied. These variables are expressed mathematically by the following equations:

$$MSE = \frac{1}{N} \sum_{i=1}^N (X_{targ(i)} - X_{ANN(i)})^2, \quad (2.28)$$

$$R = \sqrt{1 - \frac{\sum_{i=1}^N (X_{targ(i)} - X_{ANN(i)})^2}{\sum_{i=1}^N (X_{targ(i)})^2}}, \quad (2.29)$$

$$\text{Error rate (\%)} = \left[\frac{X_{targ} - X_{ANN}}{X_{targ}} \right] \times 100. \quad (2.30)$$

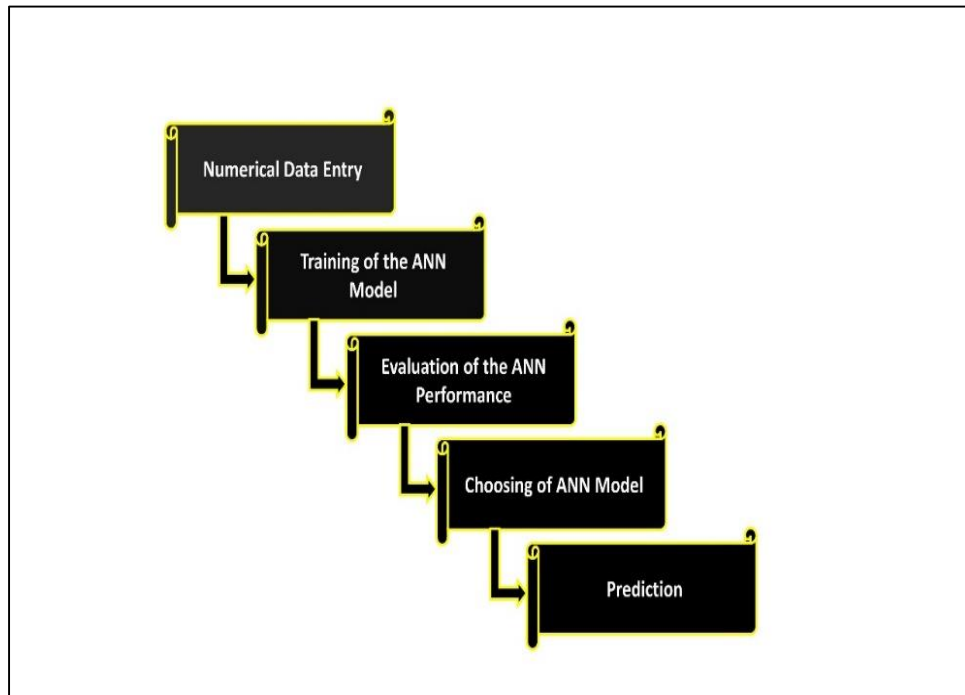


Fig. 2.2: Workflow of ANN

Further the graphical representation of ANN and numerical values such as *MSE*, training, validation and gradient are discussed in detail in result and discussion section of this chapter.

2.5 Results and Discussion

The work focuses on the flow behavior of a non-Newtonian hybrid nanofluid over a cylinder with thermal radiation embedded in a porous media. The Tiwari-Das model is used to design a flow problem over a cylinder for hybrid nanoparticles [32]. Using the Local Non-Similarity solution approach up to the third level of truncation in Eqns. (2.20)-(2.25) and the boundary condition Eqn. (2.26), the numerical solution of the boundary layer flow of the hybrid nanofluid over a cylinder is assessed. The projected solution is obtained using an ANN-based method known as MLP-ANN. An ANN-based technique called MLP-ANN is used to obtain the projected solution. The pertained parameters of interest i.e. M, Da, S_f, S_T and thermal radiations are displayed in Table 2.2.

Table 2.3 presents the numerical values of gradient, Mu and MSE for scenario 1 for different case 1-3. Performance of the MLP-ANN is obtained 1.27×10^{-10} , 1.25×10^{-10} , and 1.46×10^{-10} against epoch 332, 500, and 416 for scenario 1 of case 1-3. The numerical values of Mu and gradient for an estimated result of the current problem are $[1.0 \times 10^{-8}, 1.0 \times 10^{-8}, 1.0 \times 10^{-8}]$, and $[9.95 \times 10^{-8}, 9.95 \times 10^{-8}$ and $9.94 \times 10^{-8}]$ for scenario 1 of case 1-3 respectively. In an ANN, an error or cost functions is defined that quantifies the difference between the actual outputs and predicted outcomes using an MLP-ANN. The MSE simple measure of error. The training results of the designed ANN schemes are illustrated in Fig. 2.3(a) for Scenario 1 in case 1-3. It can be observed that the MSE values, which are high in the initial stages of the training sections, decreases with higher values of epochs. Based on the function of the MLP model, MSE values are decrease when the number of iterations is increases, and as a result, the training session of the ANN scheme vanishes as the maximum accuracy is reached. The

training set-up of the ANN schemes is depicted in Fig. 2.3(b). Gradient descent allows the optimization of the network parameters by moving in the opposite direction to the gradient of the loss function. In Fig. 2.3(b), it is observed that the gradient decreases with increasing values of epochs. The training phase was finished error-free for each scheme, and there is no error repetition. Zero is the valuation of the verification test. A histogram of the error between the desired and expected values during feed forward neural network training is called an error histogram. These flaws show how the intended outputs and the predicted outputs diverge. The error histogram of the numerical data of the suggested problem shows the difference between the expected and actual values (Fig. 2.3(c)). However, it is obtained that numerical evaluations of error presented on x-axis of plots are minor. The existing situation of data utilized in testing, training, and validation of ANN schemes are presented in Fig. 2.3(d). The y-axis shows the results of the ANN scheme, while the x-axis shows the goal numerical values. The proximity of data points representing target and predicted values is found to have very little inaccuracy. A dramatic reduction in average error is indicated by the fitted line near the zero line. A high correlation exists between an ANN's accuracy and R values that are almost equal to 1. It should be highlighted that the computed R values for each stage are almost equal to 1. Fig. 2.3(d) illustrates how the ANN is built to produce predicted values with extremely low error rates. Fig. 2.3(e) shows a visual evaluation of the performance of ANN model. When the graphs are researched, it is observed that the model-predicted output and the actual output are well matched, and the ANN model vanishes at almost zero error-line. It is also demonstrated that the optimal curve fitness function asymptotically satisfies the boundary condition. In the presence of a magnetic force in the system, a resistive force is generated, known as the Lorentz force. The resistive Lorentz force in a hybrid nanofluid causes the magnetic parameter M to start

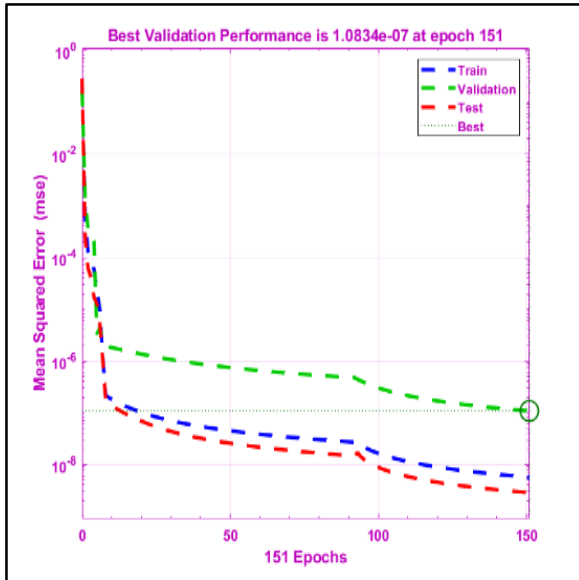
dampening the velocity. According to Fig. 2.3 (f), the non-Newtonian hybrid nanofluid's boundary layer flow velocity significantly decreased as M increased. Fig. 2.3 (g) illustrates how the magnetic parameter M affects the temperature profile. A stronger magnetic field, which resists fluid motion and produces more heat, is the outcome of a higher M value. Consequently, as M grows, the fluid's temperature rises.

Table 2. 2: Relevant parameters for the suggested model's flow analysis.

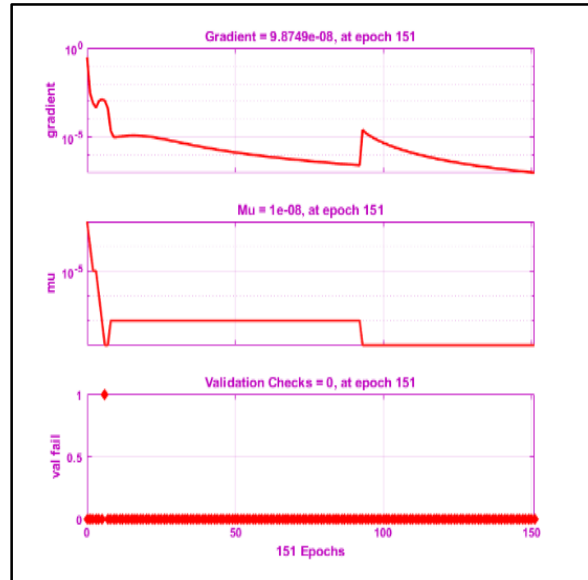
Scenario	Case	Pertained parameter				
		M	Da	Sf	S_T	Rd
1	1	0.3	0.5	0.1	0.5	3.4
	2	0.6				
	3	0.9				
2	1	0.6	0.5			
	2		0.7			
	3		0.9			
3	1			0.1		
	2			0.2		
	3			0.3		
4	1				0.5	
	2				0.7	
	3				0.9	
5	1					3.4
	2					4.4
	3					5.4

Table 2. 3: Relative assessment over backpropagation networks for scenario 1

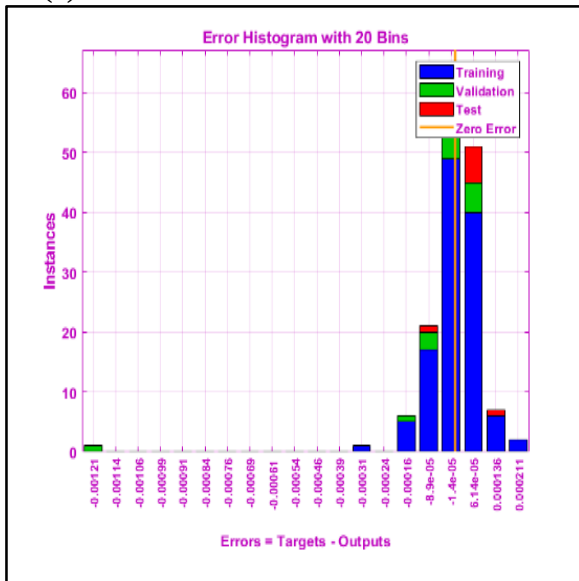
Scenario	Case	Error analysis for different level			Performance	Gradient	Mu	Epoch
		Training	Validation	Testing				
1	1	5.6834×10^{-9}	1.08339×10^{-7}	2.8634×10^{-9}	5.68×10^{-9}	9.87×10^{-8}	1×10^{-8}	151
	2	5.2823×10^{-9}	5.50863×10^{-9}	1.0798×10^{-7}	5.28×10^{-9}	9.82×10^{-9}	1×10^{-8}	146
	3	3.2678×10^{-9}	1.78245×10^{-8}	1.15376×10^{-9}	3.27×10^{-9}	9.87×10^{-9}	1×10^{-8}	108



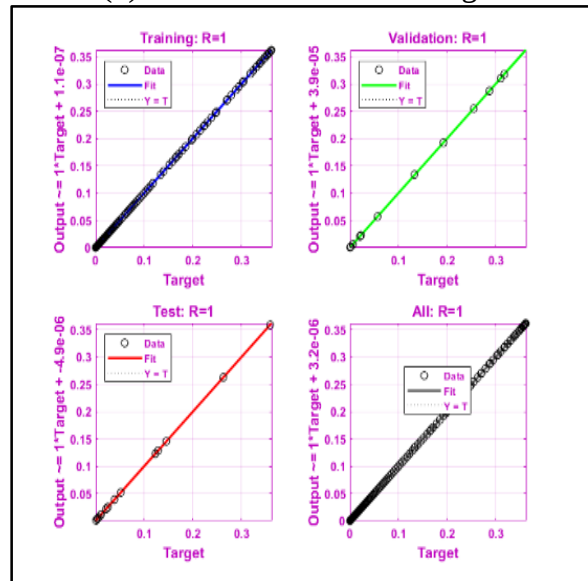
(a) Performance of ANN in terms of MSE



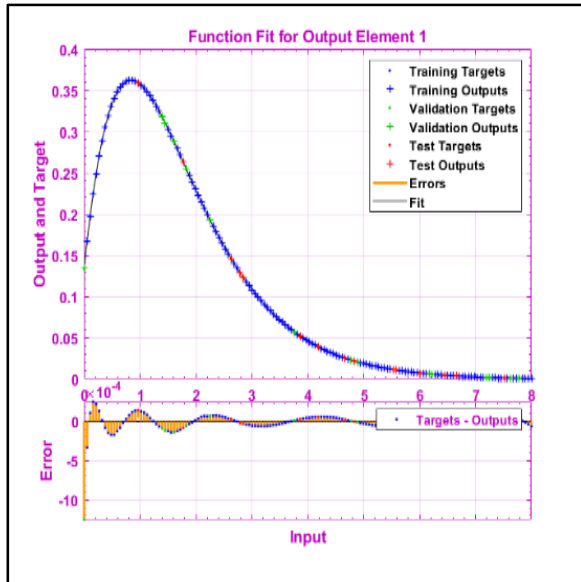
(b) Gradient of function using ANN



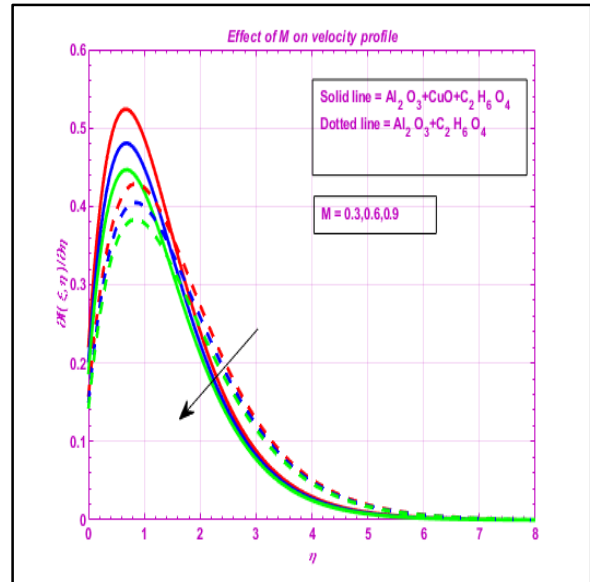
(c) Error analysis in term of Histogram



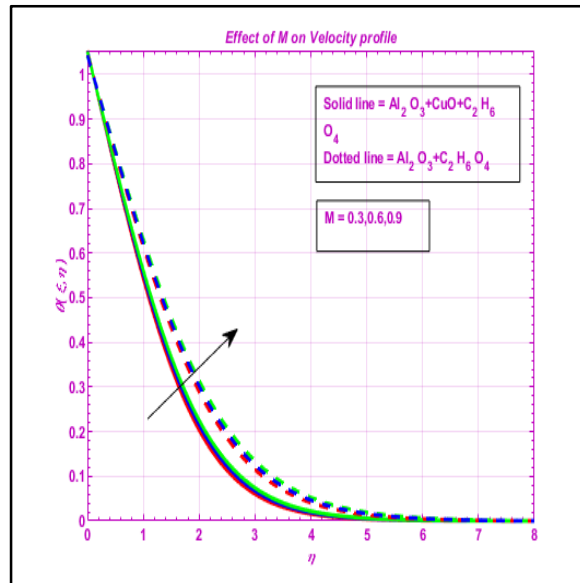
(d) Linear regression analysis with ANN



(e) Curve fitness function with ANN



(f) Effect of M on velocity profile



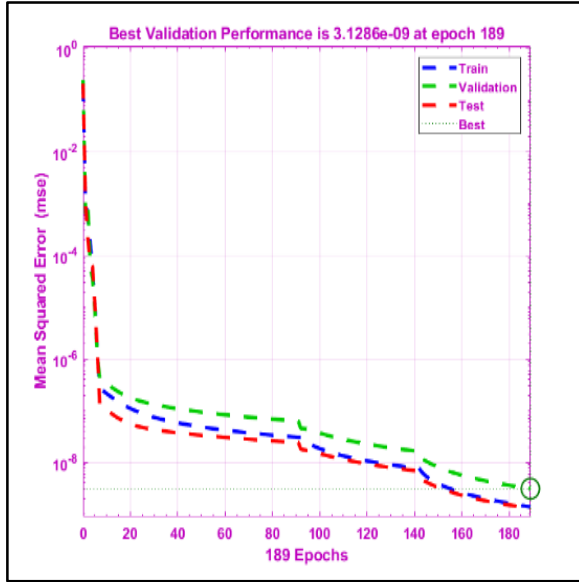
(g) Effect of M on velocity profile

Fig. 2.3: A visual representation of the data analysis for case 1 in scenarios 1.

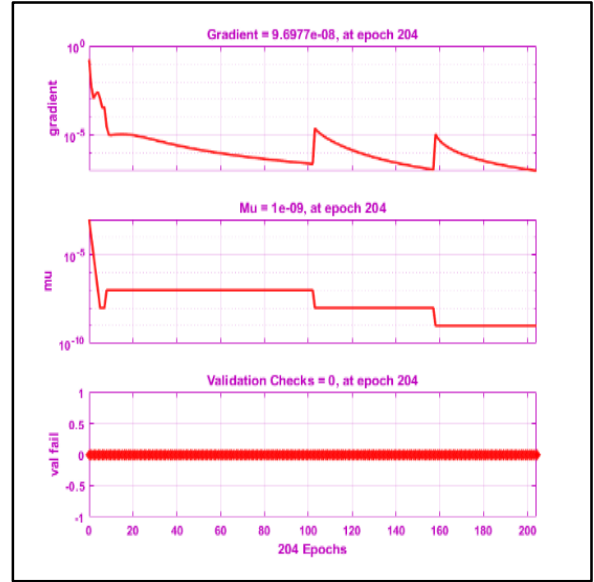
The gradient, μ , MLP ANN performance, and MSE tabulated data for cases 1-3 of scenario 2 are shown in Table 2.4. An illustration of the MSE for scenario 2 may be found in Fig. 2.4(a). The accuracy of the suggested MLP-ANN model was determined to be at 10^{-10} , as shown in Fig. 2.4(a), which shows its convergence. For Scenario 2, the gradient is represented graphically in Fig. 2.4(b). By moving in the opposite direction as the loss function's gradient, the gradient enables network parameter optimization with a convergence rate of 10^{-5} as illustrated in Fig. 2.4(b). Fig. 2.4(c) shows the error histogram, which shows the discrepancy between the fluid model's actual and anticipated values over a cylinder. It is calculated how accurate and valid the predicted solution is. Fig. 2.4(d) displays the correlation index and linear regression for Scenario 2. The best fit of the model was achieved when the correlation index value was near 1. Fig. 2.4(e) displays the curve fitness function that works well in scenario 2. Additionally, the asymptotic satisfaction of the boundary condition by the optimal curve fitness function is shown. Darcy number Da effects on velocity of the MHD boundary layer flow of a hybrid nanofluid over a cylinder is shown in Fig. 2.4(f). The fluid velocity and heat transfer study of the MHD boundary layer flow of a non-Newtonian hybrid nanofluid is significantly influenced by the Darcy number. When the Darcy number is high, the fluid velocity is less resistive due to the high permeability factor of the porous media. The velocity of the Casson hybrid nanofluid increases as the Da number grows. As the Darcy number Da increases, the fluid's temperature drops. The impact of Da on the temperature profile is depicted in Fig. 2.4(g).

Table 2. 4: Relative assessment over backpropagation networks for scenario 2 case 3.

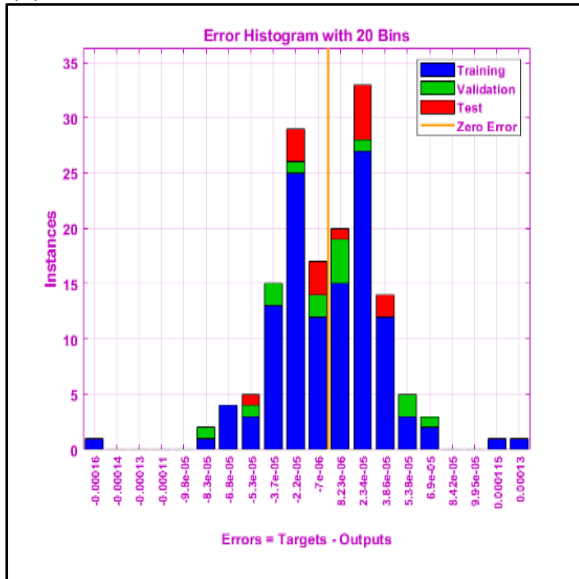
Scenario	Case	Error analysis for different level			Performance	Gradient	μ	Epoch
		Training	Validation	Testing				
2	1	1.3936×10^{-9}	3.1286×10^{-9}	1.3226×10^{-9}	1.39×10^{-9}	9.73×10^{-8}	1×10^{-9}	189
	2	5.0949×10^{-9}	5.5887×10^{-9}	5.23985×10^{-9}	5.09×10^{-9}	9.72×10^{-8}	1×10^{-8}	190
	3	4.9633×10^{-9}	5.03200×10^{-9}	1.60619×10^{-8}	4.96×10^{-9}	9.71×10^{-8}	1×10^{-8}	108



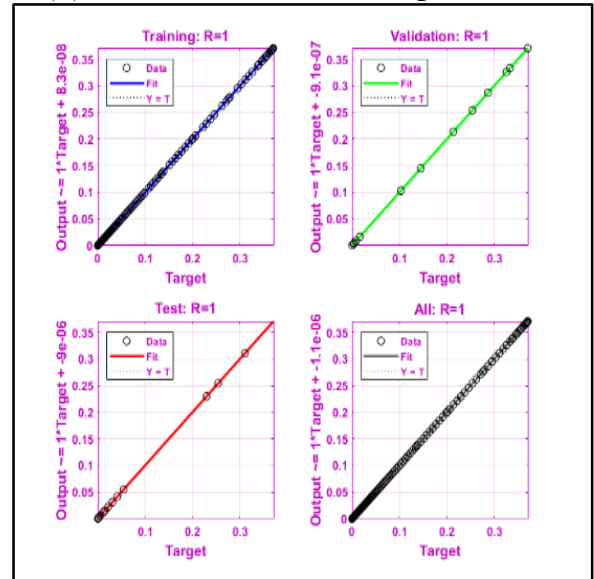
(a) Performance of ANN in terms of MSE



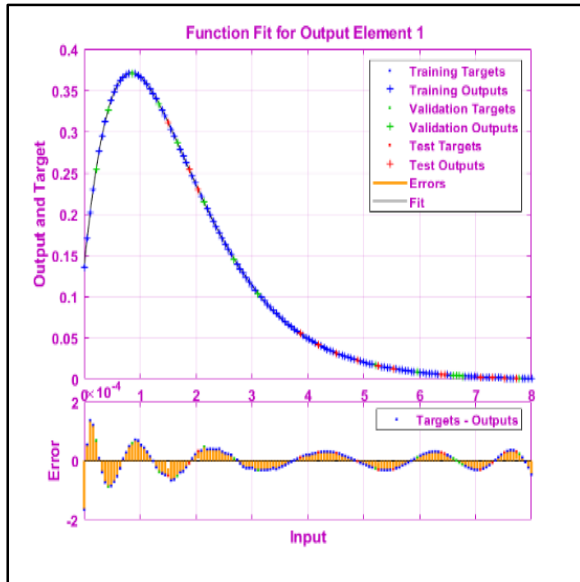
(b) Gradient of function using ANN



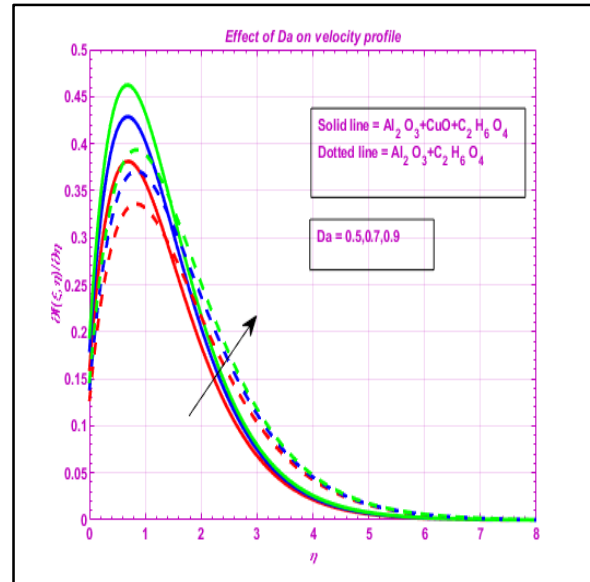
(c) Error analysis in term of Histogram



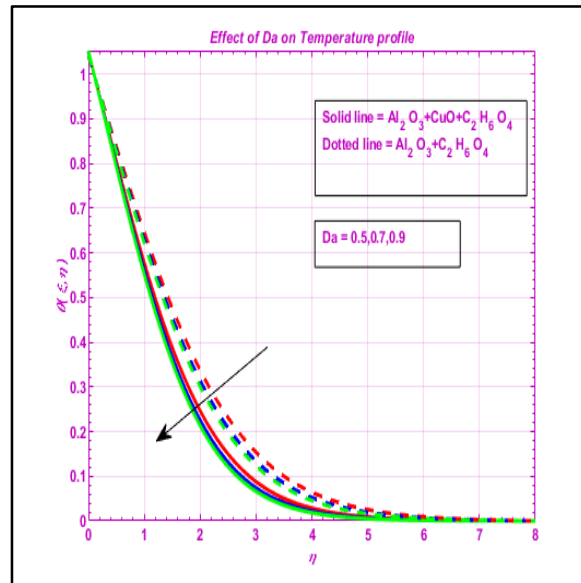
(d) Linear regression analysis with ANN.



(c) Curve fitness function with ANN



(f) Effect of Da on velocity profile



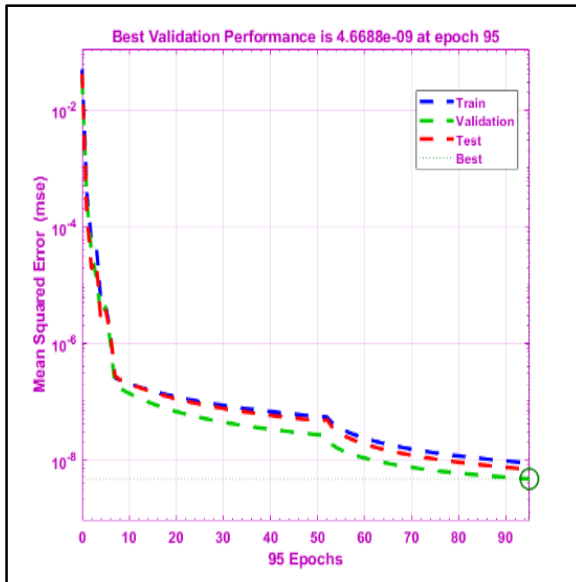
(g) Effect of Da on temperature profile

Fig. 2.4: A visual representation of the data analysis for case 1 in scenario 2.

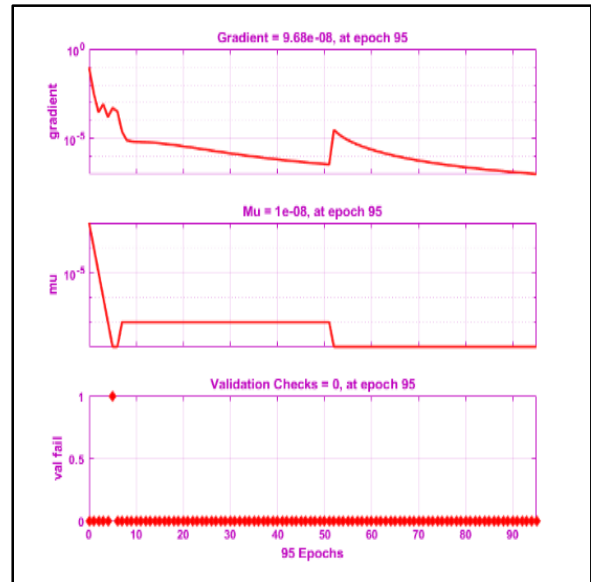
Table 2.5 presents the tabulated values of the gradient, Mu , performance of MLP ANN and MSE for case 1-3 of Scenario 3. Fig. 2.5(a) depict the graphical representation of the MSE for scenario 3. The suggested ANN model's convergence is depicted in Fig. 2.5(a), where accuracy is found to be at 10^{-8} and 10^{-10} respectively. Fig. 2.5(b) shows the convergence of the gradient curve for the predicted solution of the non-Newtonian hybrid nanofluid over a cylinder for scenario 3 respectively. The error histogram is displayed in Fig. 2.5(c) for scenario 3 respectively. The correlation index and linear regression for scenario 3 are shown in Fig. 2.5(d). The optimal curve fitness functions for scenario 3 are shown in Fig. 2.5(e). The effect of S_f on velocity profile is depicted in Fig. 2.5 (f). As slip parameter S_f increases the velocity of the fluid near the surface increases, showing an enhancement in flow due to stronger stretching of the cylinder surface. However, beyond a certain distance from the surface, the velocity gradually decrease due to higher viscous and inertial resistance. The impact of slip parameter S_f on the fluid temperature is shown in Fig. 2.5 (g). The temperature of the flow field reduces and thereby decrease the thermal boundary layer thickness as there is a growth in the value of slip parameter S_f as displayed in Fig. 2.5 (g).

Table 2. 5: Relative assessment over backpropagation networks for scenario 3

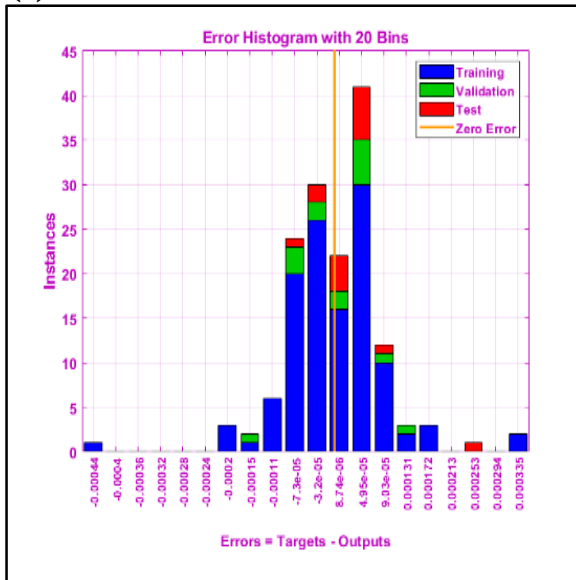
Scenario	Case	Error analysis for different level			Performance	Gradient	Mu	Epoch
		Training	Validation	Testing				
3	1	8.7602×10^{-9}	4.66882×10^{-9}	6.7907×10^{-9}	8.76×10^{-9}	9.69×10^{-8}	1×10^{-8}	95
	2	6.4703×10^{-9}	8.24242×10^{-9}	5.8979×10^{-9}	6.47×10^{-9}	9.53×10^{-8}	1×10^{-8}	108
	3	3.933×10^{-9}	6.64022×10^{-9}	6.3471×10^{-9}	3.93×10^{-9}	9.80×10^{-8}	1×10^{-8}	76



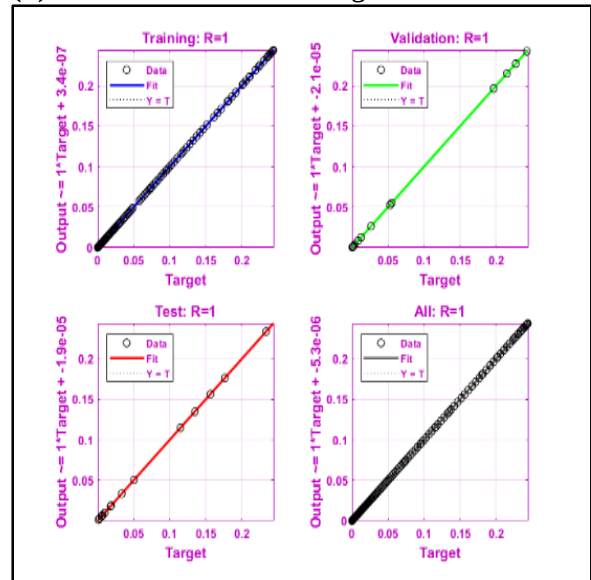
(a) Performance of ANN in terms of MSE



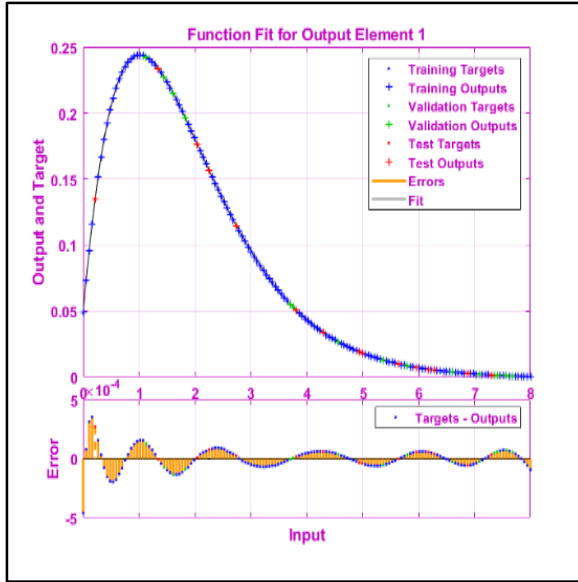
(b) Gradient of function using ANN



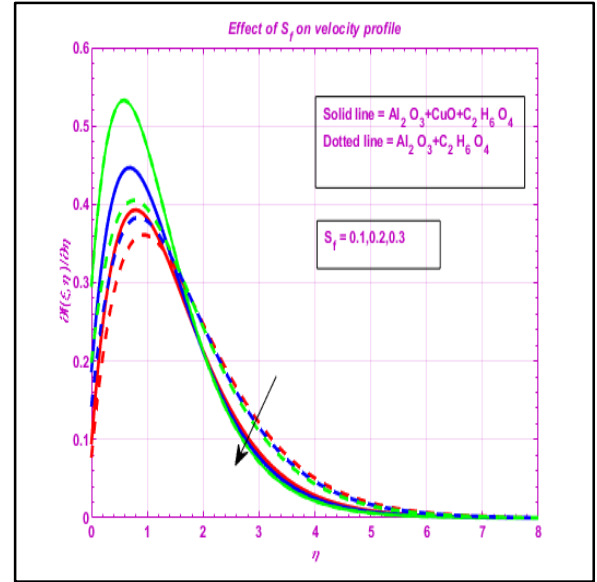
(c) Error analysis in terms of Histogram



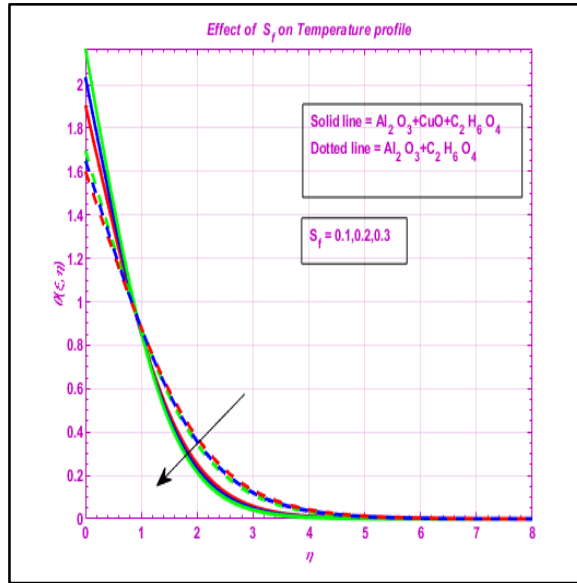
(d) Linear regression analysis with ANN



(e) Curve fitness function using ANN



(f) Effect of S_f on velocity profile



(g) Effect of S_f on temperature profile

Fig. 2.5: A visual representation of the data analysis for case 1 in scenarios 3.

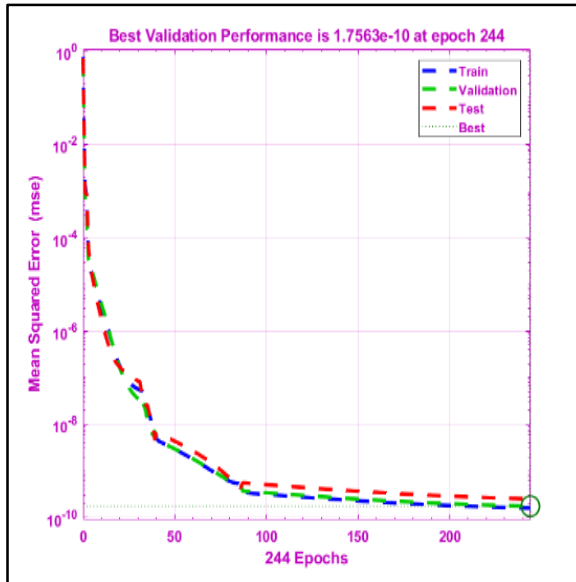
Table 2.6-2.7 presents the tabulated values of the gradient, Mu , performance of MLP ANN and MSE for case 1-3 of Scenario 4-5. Fig. 2.6(a)-2.7(a) depict the graphical representation of the MSE for scenario 4-5. The suggested ANN model's convergence is displayed in Figs. 2.6(a)–2.7(a), where accuracy is found to be at 10^{-8} and 10^{-10} correspondingly. Fig. 2.6(b)-2.7(b) show, the convergence of the gradient curve for the predicted solution of the non-Newtonian hybrid nanofluid over a cylinder for scenario 4-5 respectively. The error histogram is displayed in Fig. 2.6(c)-2.7(c) for scenario 4-5 respectively. The correlation index and linear regression for scenario 4-5 are shown in Fig. 2.6(d)-2.7(d). The optimal curve fitness functions for scenario 4-5 are shown in Fig. 2.6(e)-2.7(e). The effects of the thermal jump parameter S_T are shown in Fig. 2.6 (f). When the effects S_T get improve, the interaction between the wall and fluid become relatively weak, and as a result, the velocity near the surface decrease. Fig.2.6 (g) shows the effects of thermal jump parameter S_T on the thermal boundary layer of a Casson hybrid nanofluid. It should be noted that as the thermal jump S_T increases, then thermal boundary layer decrease. Additionally, it is examine that the thermal slip parameter has its greatest impact close to the surface wall surface. Figures 2.7(f) and 2.7(g) show how the radiation parameter R_d affects the velocity and temperature profile. The thermal boundary layer's thickness rises in tandem with the radiation parameter. This means that a greater proportion of the fluid is undergoing the heating effect, which raises the temperature and causes the velocity to rise in tandem.

Table 2. 6: Relative assessment over backpropagation networks for scenario 4

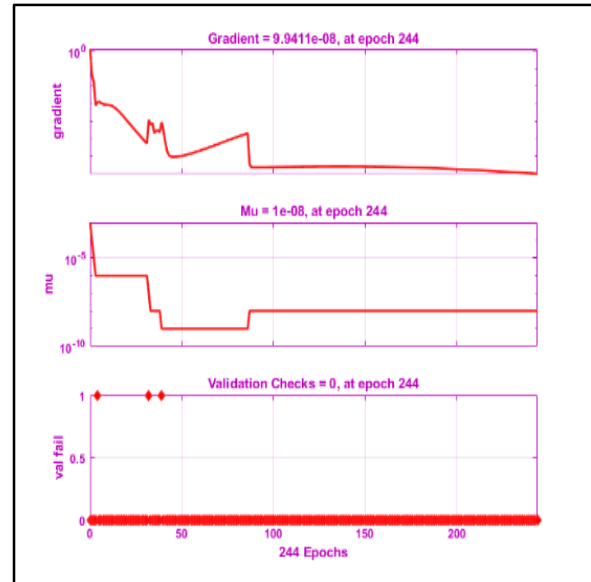
Scen ario	Case	Error analysis for different level			Performance	Gradient	<i>Mu</i>	Epoch
		Training	Validation	Testing				
	1	1.5996×10^{-10}	1.75631×10^{-10}	2.5186×10^{-10}	1.60×10^{-10}	9.94×10^{-8}	1×10^{-8}	244
4	2	1.0187×10^{-10}	5.68581×10^{-11}	1.7961×10^{-10}	1.02×10^{-10}	9.50×10^{-8}	1×10^{-9}	193
	3	2.7781×10^{-10}	2.11341×10^{-10}	3.0402×10^{-10}	2.78×10^{-10}	9.99×10^{-8}	1×10^{-8}	260

Table 2. 7: Relative assessment over backpropagation networks for scenario 5

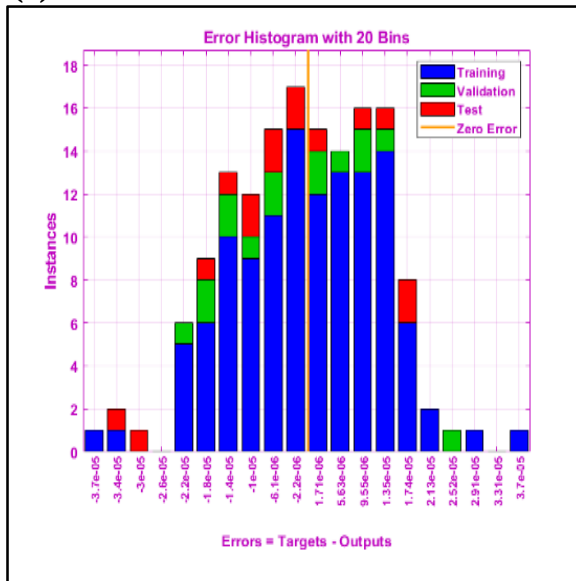
Scen ario	Case	Error analysis for different level			Performance	Gradient	<i>Mu</i>	Epoch
		Training	Validation	Testing				
	1	7.1889×10^{-10}	6.35115×10^{-10}	7.6243×10^{-10}	7.19×10^{-10}	9.67×10^{-8}	1×10^{-8}	189
5	2	4.584×10^{-10}	6.73473×10^{-10}	3.1610×10^{-10}	4.58×10^{-10}	9.72×10^{-8}	1×10^{-8}	242
	3	1.6488×10^{-10}	1.03590×10^{-10}	1.607×10^{-10}	1.65×10^{-10}	9.92×10^{-8}	1×10^{-8}	146



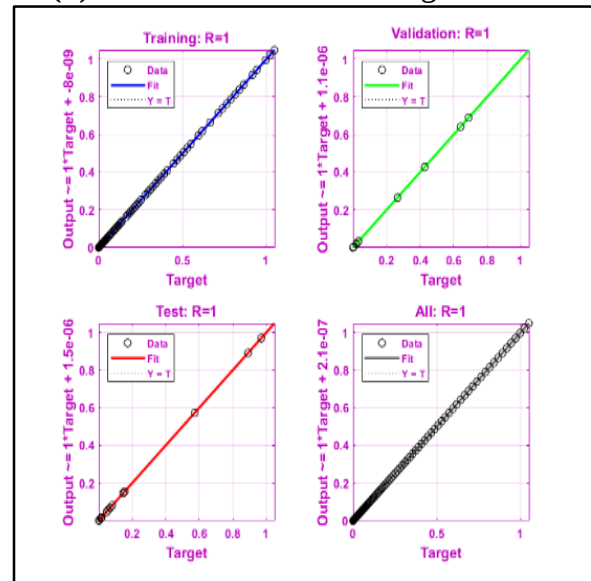
(a) Performance of ANN in terms of MSE



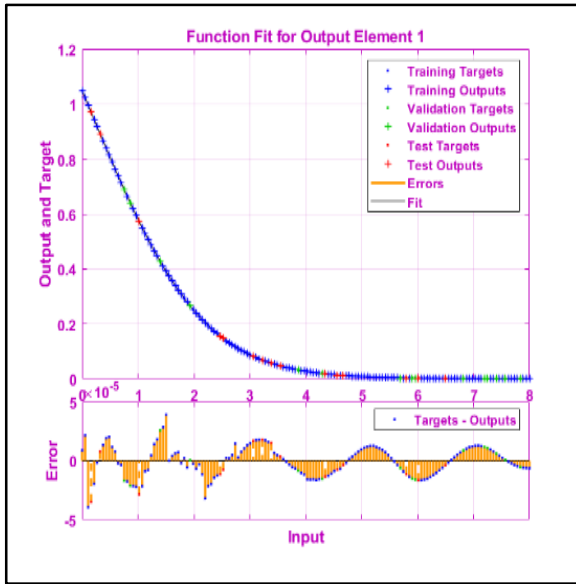
(b) Gradient of function using ANN



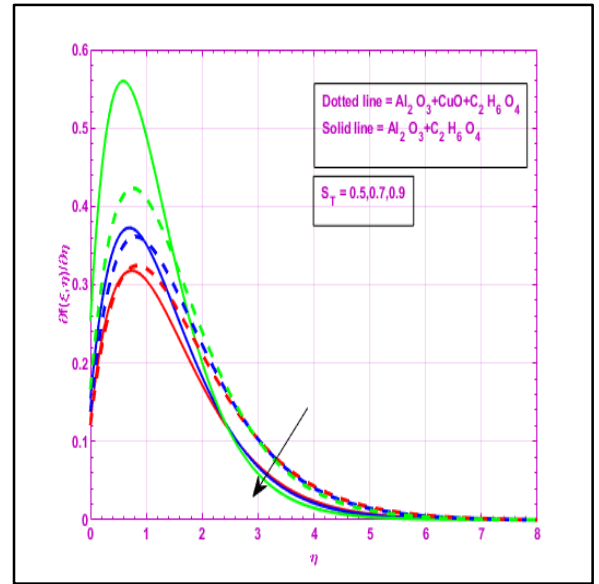
(c) Error analysis in terms of Histogram



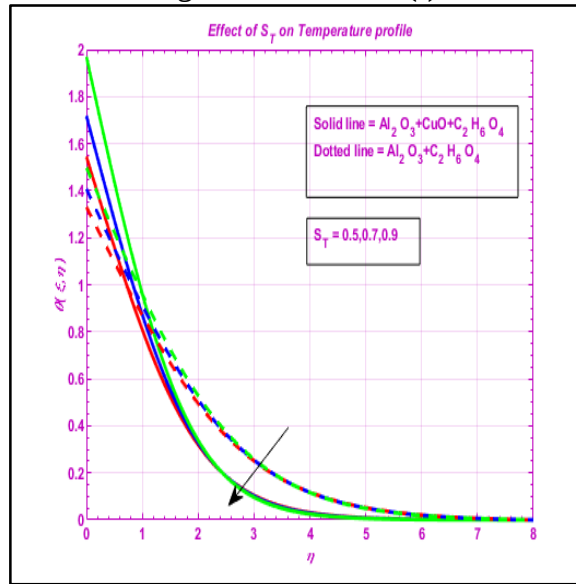
(d) Linear regression analysis with ANN



(e) Curve fitness function using ANN

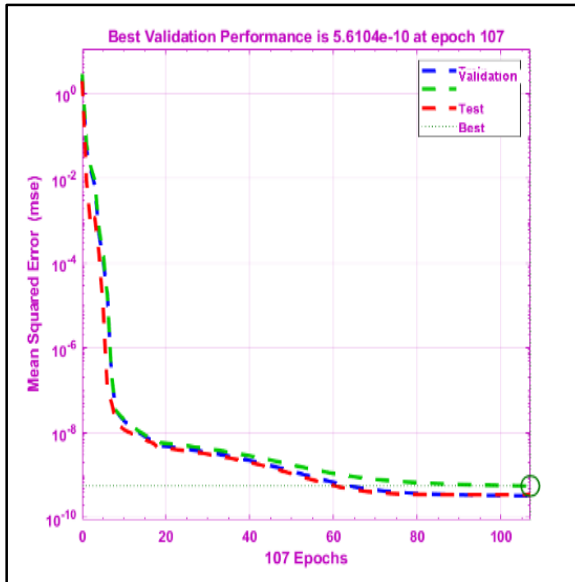


(f) Effect of S_T on velocity profile

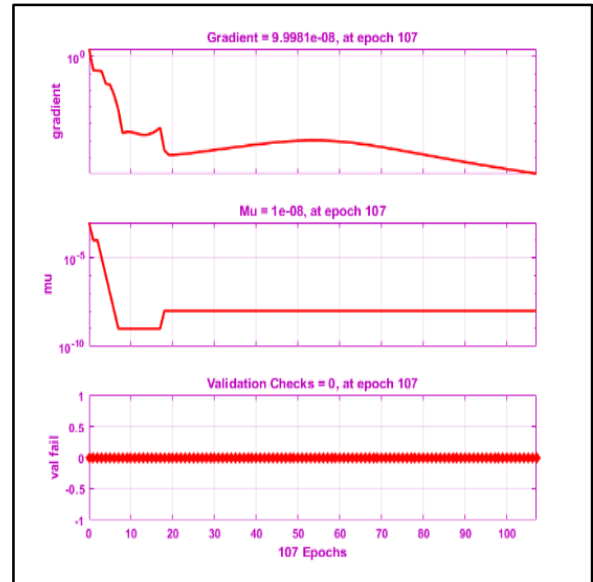


(g) Effect of S_T on velocity profile

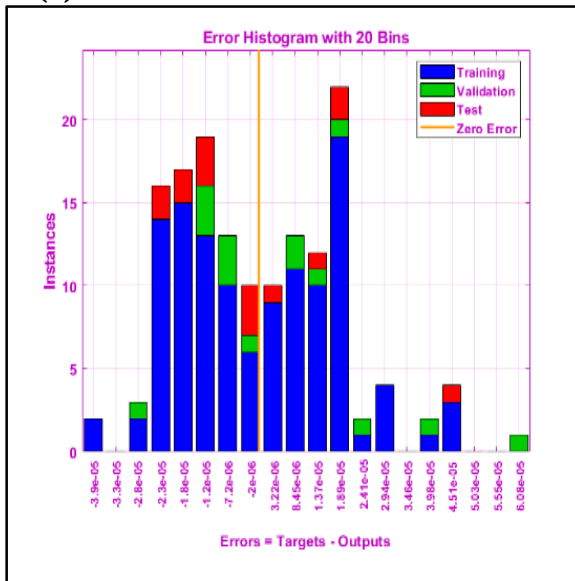
Fig. 2.6: A visual representation of the data analysis for case 1 in scenarios 4.



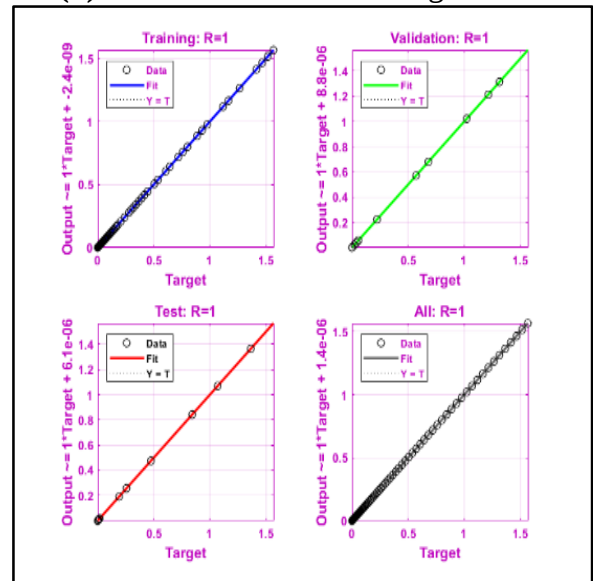
(a) Performance of ANN in terms of MSE



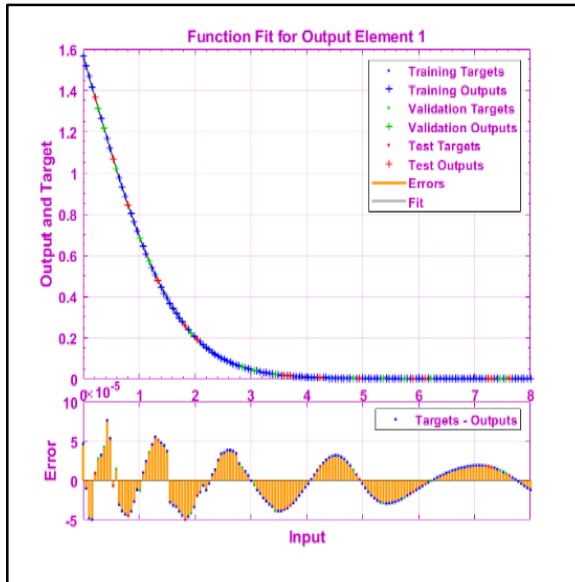
(b) Gradient of function using ANN



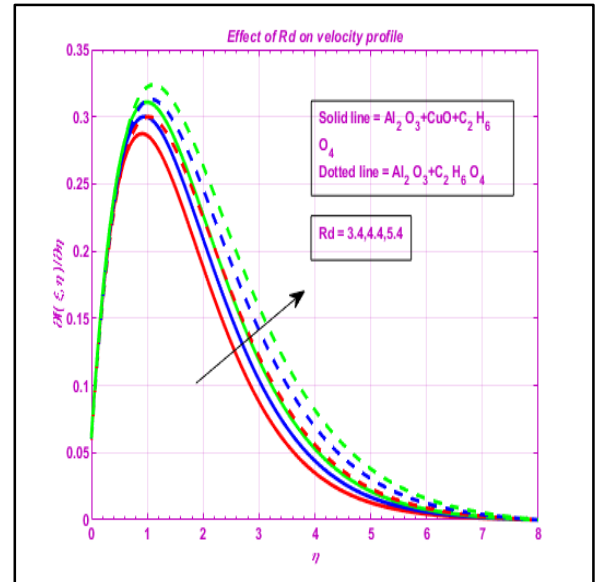
(c) Error analysis in terms of Histogram



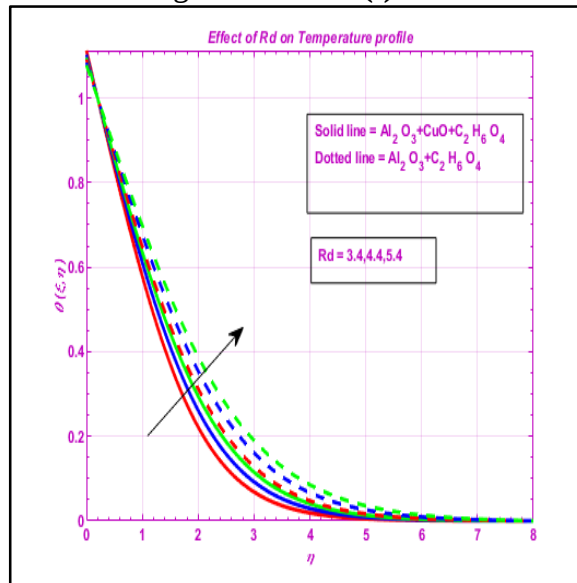
(d) Linear regression analysis with ANN



(e) Curve fitness function using ANN



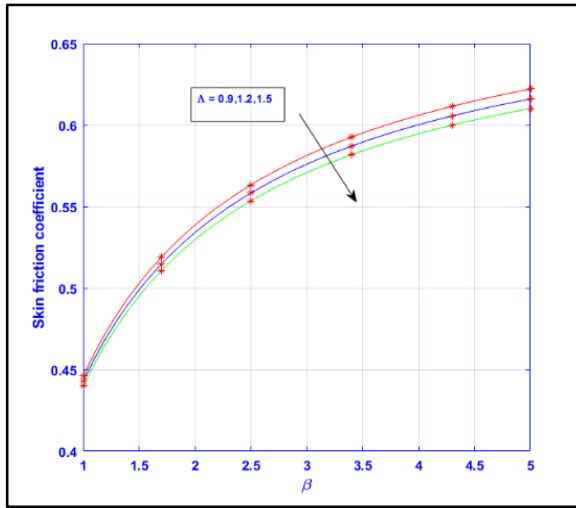
(f) Effect of Rd on velocity profile



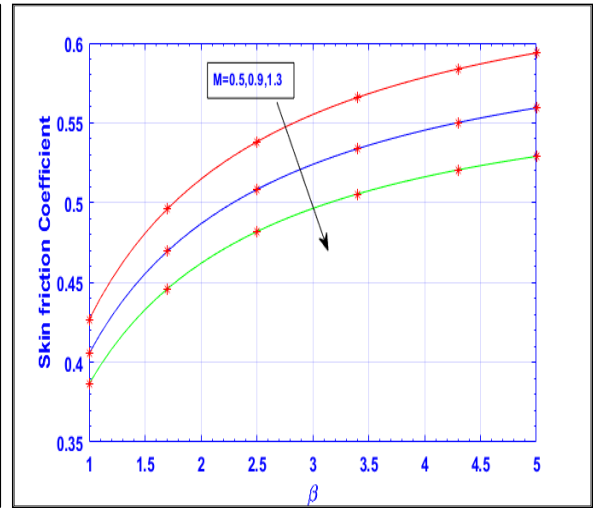
(g) Effect of Rd on temperature.

Fig. 2.7: A visual representation of the data analysis for case 1 in scenarios 5.

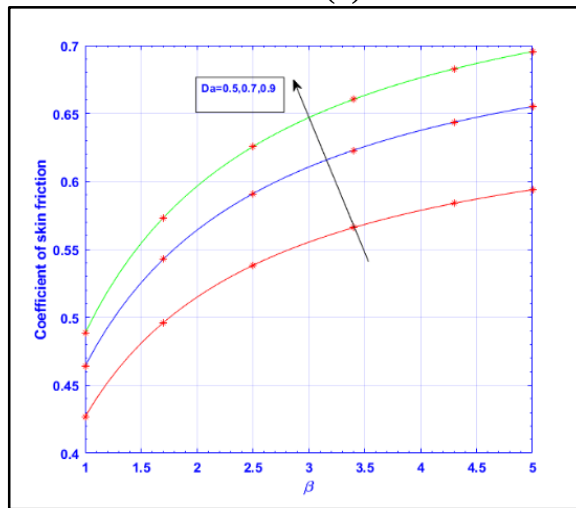
The impact of magnetic parameter M on the coefficient of skin friction is depicted in Fig. 2.8 (a). As the magnetic field intensity increases, the Lorentz force generation becomes stronger, which causes a decrease in velocity. The fluid motion near the surface of the circular cylinder decreases as a result of the Lorentz force acting against the direction of fluid flow. Taking into account that fluid speed and magnetic field intensity are inversely correlated, which lowers the skin friction coefficient. The impact of the Forchheimer parameter Λ on the skin friction coefficient is seen in Fig. 2.8 (b). The inertia effect intensifies as the flow slows down on the cylinder surface as the Forchheimer parameter rises. Figure 2.8 (c) illustrates how the Darcy number Da affects the skin friction coefficient and shows that the skin friction coefficient C_f tends to increase in tandem with the Darcy number's growth. Fig. 2.8 (d) shows how the Radiation parameter Rd affects the Nusselt number. The findings show that the local Nusselt number decreases as the radiation parameter Rd increases. This process is caused by the boundary layer's radiative heat transport being stronger, which lowers the temperature gradient at the solid-fluid interface. The impact of Prandtl number Pr on the Nusselt number coefficient is shown in Fig. 2.8 (e). The results demonstrates that a higher Prandtl number Pr decreases the thermal diffusivity of the fluid and hence makes the thermal boundary layer thinner near the cylinder surface. This increases the temperature gradient at the wall, which increases the local Nusselt number.



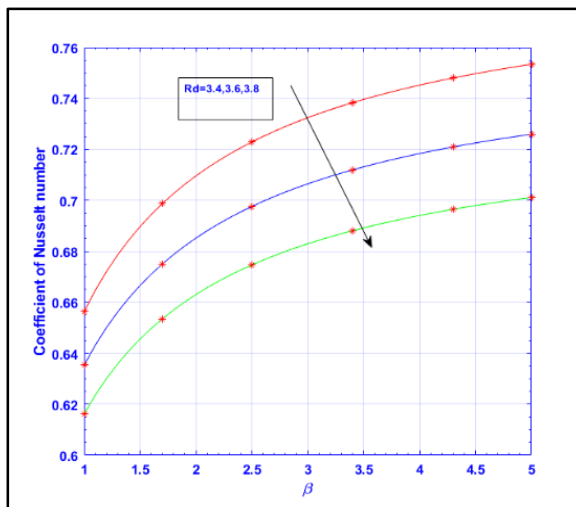
(a): Effect of Λ on Skin friction Coefficient



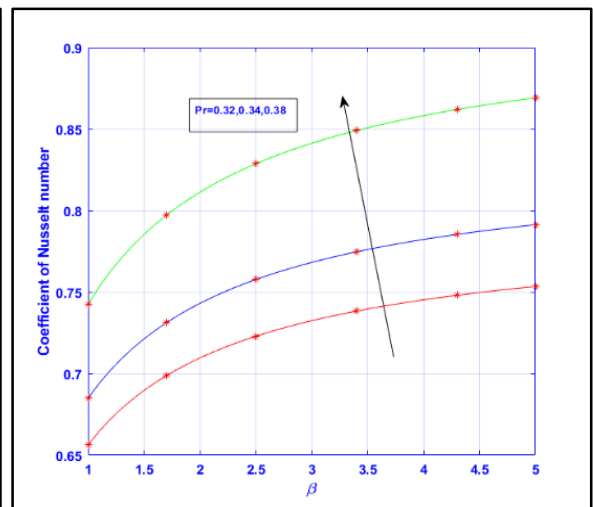
(b): Effect of M on Skin friction Coefficient



(c): Effect of Da on Skin friction Coefficient



(d): Effect of Rd on Coefficient of Nusselt number



(e): Effect of Pr on Coefficient of Nusselt number

Fig. 2.8: Effects of various parameters on the Nusselt number and skin friction coefficient.

2.6 Conclusion

An Artificial Neural Network (ANN) is used in this work to examine boundary layer flow and heat transfer enhancement in a Casson hybrid nanofluid over a cylinder, taking into account partial slip, non-Darcy porous medium, and thermal radiation effects. The following are the main findings from the analysis:

Velocity profile against the dimensionless parameter decrease with increasing M, β, S_f and S_T parameters, whereas velocity profile increases with increasing Da parameters. Temperature profile is increases with increasing dimensionless parameter Rd and decreases with increasing pr, S_f and S_T .

Skin friction coefficient decrease with increasing magnetic parameter M and Forchhiemeter parameter Λ , while decreasing with increasing darcy parameter Da .

Coefficient of Nusselt number decrease with increasing radiation parameter Rd while increasing with higher value of prandlt parameter Pr .

A very small error margin was found when comparing the calculated results with the output anticipated by the ANN and the supervised machine learning method.

The proposed ANN model is considered reliable for of its high accuracy, which is consistent across training, testing, and validation when compared to the computational techniques.

Chapter 3

Physics Informed Neural Network Simulation of Non-Newtonian Flow around Cylinder

3.1 Introduction

In the domain of artificial intelligence and machine learning, physics informed neural networks become hot topic due to its wide range of application in flow problem. Physics Informed Neural Network (PINN) are a successful approach for identifying the hidden physics underlying transport phenomena through training on big set of data. This work addresses non-Newtonian Casson fluid flow over cylinder with magnetic effect through porous medium. The models are constructed and trained using TensorFlow, and the predicted solution that are generated are compared to those derived through bvp4c techniques.

3.2 Problem Formulation

The geometry and flow physics of the problem are discussed in section 2.2. The governing boundary layer equation are follow as [31]

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0, \quad (3.1)$$

$$u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = \nu \left(1 + \frac{1}{\beta} \right) \frac{\partial^2 u}{\partial y^2} - \frac{\alpha B_0^2}{\rho} u - \frac{\nu}{K} u - cu^2 + g\beta_1(T - T_\infty) \sin\left(\frac{x}{a}\right), \quad (3.2)$$

$$u \frac{\partial T}{\partial x} + v \frac{\partial T}{\partial y} = \alpha \frac{\partial^2 T}{\partial y^2} - \frac{\partial q_x}{\partial y}. \quad (3.3)$$

Whereas thermal radiation is denoted by

$$\frac{\partial q_x}{\partial y} = -\frac{16\sigma^* T_\infty^3}{3k^*} \left(\frac{\partial^2 T}{\partial y^2} \right).$$

The no slip boundary condition are as follow [2]

$$\text{At } y = 0, u = 0, v = 0, T = T_w$$

$$\text{As } y \rightarrow \infty, u \rightarrow 0, T \rightarrow T_\infty. \quad (3.4)$$

Stream function ψ is defined by $u = \frac{\partial \psi}{\partial y}$ and $v = -\frac{\partial \psi}{\partial x}$. The dimensionless variable are expressed as follow [31]

$$\xi = \frac{x}{a}, \eta = \frac{y}{a} \sqrt[4]{Gr}, f(\xi, \eta) = \frac{\psi}{\nu \xi \sqrt[4]{Gr}}, \theta(\xi, \eta) = \frac{T - T_\infty}{T_w - T_\infty}, Gr = \frac{g \beta_1 (T_w - T_\infty) a^3}{\nu^2} \quad (3.5)$$

Applying non-similarity transformation, Eqn. (3.1) i.e. continuity equation satisfied identically, however of Eqns. (3.2)-(3.3) are obtained as follow;

$$\left(1 + \frac{1}{\beta}\right) f''' + f f'' - (1 + \xi \Lambda) f'^2 - \left(M + \frac{1}{Da}\right) f' + \frac{\sin \xi}{\xi} [\theta] = \xi \left(f' \frac{\partial f'}{\partial \xi} - f'' \frac{\partial f}{\partial \xi}\right), \quad (3.6)$$

$$\frac{1}{Pr} \left[1 + \frac{4}{3} Rd\right] \theta'' + f \theta' = \xi \left(f' \frac{\partial \theta}{\partial \xi} - \frac{\theta' \partial f}{\partial \xi}\right) \quad (3.7)$$

Dimensionless boundary conditions are obtained as follow

$$\text{at } \eta = 0, f' = 0, f = 0, \theta = 1 \quad (3.8)$$

$$\text{As } \eta \rightarrow \infty, f' \rightarrow 0, \theta \rightarrow 0$$

The physical parameter that arises in Eqns. (3.6)-(3.8) are discussed in section 2.2.

3.3 Solution of the Problem

To numerically solve the considered problem, we use the local non similarity solution method as discussed in section 2 as:

3.3.1 First Level Truncation:

By using first level truncation, the terms on the right side of Eqns. (3.6)-(3.7) are disregarded, presuming that the terms involved $\xi \frac{\partial(\cdot)}{\partial \xi}$ are minimal. The system of Equations (3.6)–(3.7) is true when $\xi \ll 1$.

$$\left(1 + \frac{1}{\beta}\right) f''' + f f'' - (1 + \xi \Lambda) f'^2 - \left(M + \frac{1}{Da}\right) f' + \frac{\sin \xi}{\xi} [\theta] = 0 \quad (3.9)$$

$$\frac{1}{pr} \left[1 + \frac{4}{3} Rd \right] \theta'' + f \theta' = 0, \quad (3.10)$$

Boundary condition for first level

$$\text{at } \eta = 0, f' = 0, f = 0, \theta = 1, \quad (3.11)$$

$$\text{As } \eta \rightarrow \infty, f' \rightarrow 0, \theta \rightarrow 0.$$

3.3.2 Second Level Truncation

The following terms are defined in order to construct the equation of higher order truncation:

$$g = \frac{\partial f}{\partial \xi}, \quad h = \frac{\partial g}{\partial \xi}, \quad \phi = \frac{\partial \theta}{\partial \xi}, \quad \chi = \frac{\partial \phi}{\partial \xi} \quad (3.12)$$

The derivative of equations (6-7) and the boundary condition Eqn. (8) with respect to ξ yields the subsidiary equations for g and ϕ and their boundary condition.

$$\left(1 + \frac{1}{\beta}\right) f''' + f f'' - (1 + \xi \Lambda) f'^2 - \left(M + \frac{1}{Da}\right) f' + \frac{\sin \xi}{\xi} (\theta) = \xi (f' g' - f'' g), \quad (3.13)$$

$$\frac{1}{pr} \left[1 + \frac{4}{3} Rd \right] \theta'' + f \theta' = \xi (f' \theta - \theta' g), \quad (3.14)$$

$$\left(1 + \frac{1}{\beta}\right) g''' + f g'' + 2 f'' g - f' g' - (1 + \xi \Lambda) g'^2 - \Lambda f' - \left(M + \frac{1}{Da}\right) g' + \quad (3.15)$$

$$\left[\left(\frac{\sin \xi}{\xi} \phi \right) \left(\frac{\xi \cos(\xi) - \sin(\xi)}{\xi^2} \right) \theta \right] = \xi (g' g' - g'' g),$$

$$\frac{1}{pr} \left[1 + \frac{4}{3} Rd \right] \phi'' + f \phi' - f' \phi + 2 \theta' g = \xi (g' \phi - \phi' g), \quad (3.16)$$

The boundary conditions are describe as;

$$\begin{aligned} \text{at } \eta = 0, \quad f' = 0, \quad f = 0, \quad \theta = 1, \quad g' = 0, \\ g = 0, \quad \phi = 0, \end{aligned} \quad (3.17)$$

$$\text{As } \eta \rightarrow \infty, f' \rightarrow 0, \theta \rightarrow 0, g' \rightarrow 0, \phi \rightarrow 0.$$

3.3.3 Third Level Truncation:

At the third truncation level, the conservation equations for the f and θ functions, as well as their corresponding equation for g and ϕ , are maintained without approximation. The first derivative of Eqns. (3.13)–(3.14) is used to define boundary conditions and other subsidiary equations for h and χ . The equation system for the third level of truncation is obtained by eliminating terms that involve $\frac{\partial h}{\partial \xi}$, $\xi \frac{\partial h'}{\partial \xi}$ and $\xi \frac{\partial \chi}{\partial \xi}$.

$$\left(1 + \frac{1}{\beta}\right) f''' + f f'' - (1 + \xi \Lambda) f'^2 - \left(M + \frac{1}{Da}\right) f' + \frac{\sin \xi}{\xi} \theta = \xi (f' g' - f'' g), \quad (3.18)$$

$$\frac{1}{pr} \left[1 + \frac{4}{3} Rd\right] \theta'' + f \theta' = \xi (f' \theta - \theta' g), \quad (3.19)$$

$$\left(1 + \frac{1}{\beta}\right) g''' + f g'' + 2 f'' g - f' g' - (1 + \xi \Lambda) g'^2 - \Lambda f' - \left(M + \frac{1}{Da}\right) g' + \left[\left(\frac{\sin \xi}{\xi} \phi\right) \left(\frac{\xi \cos(\xi) - \sin(\xi)}{\xi^2}\right) \theta\right] = \xi (g' g' - g'' g + f' h' - f'' h), \quad (3.20)$$

$$\frac{1}{pr} \left[1 + \frac{4}{3} Rd\right] \phi'' + f \phi' - f' \phi + 2 \theta' g = \xi (g' \phi - \phi' g + f' \chi - \theta' h), \quad (3.21)$$

$$\begin{aligned} &\left(1 + \frac{1}{\beta}\right) h''' + f h'' + 3 f'' h + 4 g'' g - 2 f' h' - 2 g' g' - (1 + \xi \Lambda) h'^2 - \\ &2 \Lambda g'^2 - \left(M + \frac{1}{Da}\right) h' + 2 \left(\frac{\xi \cos(\xi) - \sin(\xi)}{\xi^2}\right) \phi + \frac{\sin \xi}{\xi} \chi - \\ &\left(\frac{\xi^2 \sin \xi + 2 \xi \cos \xi - 2 \sin \xi}{\xi^3}\right) \theta = \xi (3 h' g' - h'' g - 2 g'' h), \end{aligned} \quad (3.22)$$

$$\begin{aligned} &\frac{1}{pr} \left[1 + \frac{4}{3} Rd\right] \chi'' - 2 \phi' g + f \chi' - 2 g' \phi - 2 f' \chi + \theta' h = \xi (2 g' \chi - \chi' g + \\ &h' \phi - 2 \phi' h). \end{aligned} \quad (3.23)$$

Boundary condition for third level of truncation are given below;

$$\begin{aligned} &\text{at } \eta = 0, \quad f' = 0, \quad f = 0, \quad \theta = 1, \quad g' = 0, \\ &g = 0, \quad \phi = 0, \quad h = 0, \quad h' = 0, \quad \chi = 0, \end{aligned} \quad (3.24)$$

As $\eta \rightarrow \infty, f' \rightarrow 0, \theta \rightarrow 0, g' \rightarrow 0, \phi \rightarrow 0, h' \rightarrow 0, \chi \rightarrow 0$.

Table 3.1: Comparison of Numerical results of 1st, 2nd and 3rd level truncation

		1 st Level	Truncation	2 nd Level	Truncation	3 rd Level	Truncation
M	η	$f'(\eta)$	$\theta(\eta)$	$f'(\eta)$	$\theta(\eta)$	$f'(\eta)$	$\theta(\eta)$
1.5	0	0	1	0	1	0	1
1.5	1	0.747233	0.504009	0.684976	0.572098	0.648984	0.590136
1.5	2	0.509204	0.191026	0.344373	0.257344	0.362810	0.260877
1.5	3	0.229760	0.054301	0.119951	0.089297	0.140155	0.093915
1.5	4	0.090951	0.014795	0.040461	0.029879	0.046481	0.030022
1.5	5	0.030818	0.003560	0.012079	0.009042	0.014496	0.009359
1.5	6	0.009585	0.000813	0.003234	0.002582	0.004074	0.002753
1.5	7	0.002689	0.00017	0.000690	0.000065	0.000839	0.000645
1.5	8	0	0	0	0	0	0

3.4 Physics Informed Neural Network

A neural network's input layer, hidden layer, and output layer are a sequence of parallel layers that make up its fundamental architecture. Fully Connected Networks (FCN) are designed to collect data, process it, and then provide output. By dividing the one-dimensional space in direction η into discrete nodes, we begin the fundamental procedures of implementing PINN to solve the following equations. The use of PINN requires consideration of a finite range for η . Two distinct node types are shown in Fig. 3.2: the green nodes, also known as collocated nodes, and the blue nodes, also known as boundary nodes. When the boundary condition is applied at zero and infinity, the first and last nodes are assessed as boundary nodes. The intermediate nodes, which are considered to be collocated nodes scattered along the η -axis, are in charge of obeying nodes or physical laws at the positions between η_0 and η_∞ .

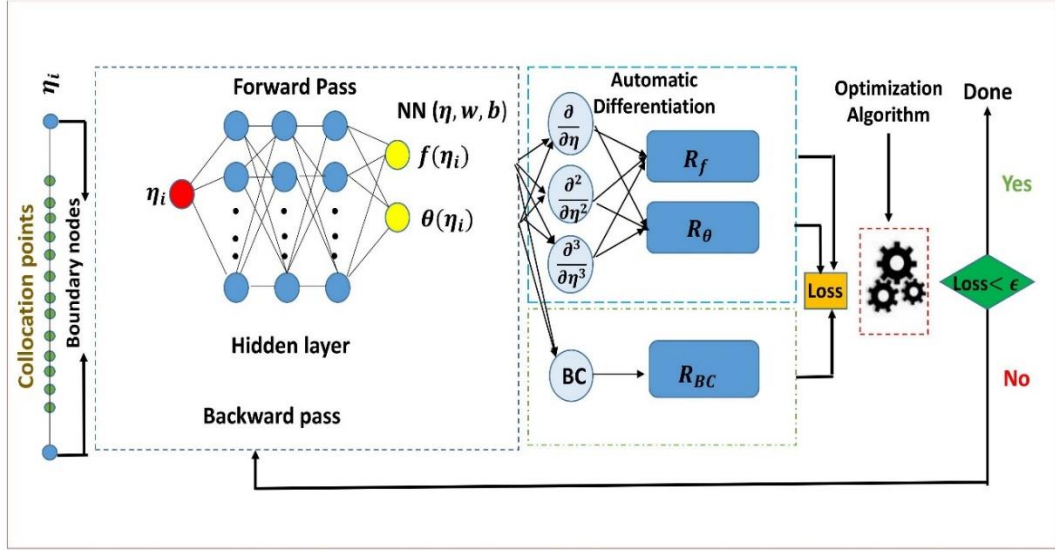


Fig. 3.1 : PINN Architecture

The first layer in Fig. 3.2 is referred as input layer that keep the input data (η_i). The next is to setup a function that maps the input data $x_i = \eta_i$ to the output $\hat{y}_i = [f(\eta_i), \theta(\eta_i)]$. The function $\hat{y}_i = F(x_i, w, b)$ involve some learning parameter that is known as weight and bias. These learning parameters need to be trained to get suitable approximation function, such that $\hat{y}_i$ appropriately expect preferred value y_i (sample output data). The difference between (y_i) and ($\hat{y}_i$) are calculated by a function termed as Loss function. The loss function can be calculated in a mean square error format, defined as,

$$\text{Loss} = \frac{\left[\sum_{i=0}^{i=m} (y_i - \hat{y}_i)^2 \right]}{m} \quad (3.25)$$

Network structure illustrated in Fig. 3.2 indicates the input data η_i supplied to the network. The first layer of inputs has single neuron storing given data. Thereafter there are three number of hidden layers containing arbitrary number of neuron and then the output layer illustrated in Fig. 3.2 consists of two neurons. The number of output layer neurons varies based on the unknown functions of the provided problem. There is a link connecting each neuron in the provided architecture in which the data flows between neurons. There is a weight for every link, which reduces or amplifies the data from the

origin node to the destination nodes. The weight analyzes the extent to which each node contributes to the results of the output. Weighted inputs are added together and a bias term is included for the calculation of participation of all nodes on predicted nodes.

$$z_i = b_i + \left(\sum_{j=1}^{j=n} a_j \cdot w_{i,j}^k \right) \quad (3.26)$$

The bias term (b_i) is added to only input and hidden layers and it has a value of one. z_i is a conventional regression analysis that is manifested as $f(x) = \alpha x + \beta$, where α and β are corresponding to weight and biases in the neural network simulations. After that an activation function is used which restricted the output in particular range. In current study the tangent hyperbolic function is utilized and the resulted z_i is passes through the given activation function.

$$\text{Output}(i) = \tanh(z_i) \quad (3.27)$$

To compute the output(z_i), the information passes through forward direction between each neuron. The process is preceded layer by layer until the estimated value $\hat{y}_i$ is computed. Since the process is started by the guessed value of weight and biases so it is presumed to see meaningful deviation in between the $\hat{y}_i$ and y_i . The value of weight and biases should be adjusted to reduce the computed loss function.

Iteratively adjusting the weights and biases to lower the loss function is the optimization process. This is accomplished by employing the backpropagation method and the chain rule to compute gradients of the loss with respect to the network parameters. These updates are performed in this study using the Adaptive Moment Estimation (Adam) optimizer. The optimization process keeps going until the network has sufficiently learned the patterns present in the data and the loss function satisfies the given convergence criteria. A paradigm for creating loss functions with ordinary differential equations (ODEs) was presented by Raissi et al. [28–29]. The loss quantifies the degree

to which the expected outputs $\hat{y}_i$ satisfy the underlying physical principles. In this work, we apply this approach to examine the boundary layer flow and heat transfer of a Casson fluid over a horizontal cylinder, taking into account the effects of a magnetic field and a porous medium. The governing equations are then non-dimensionalized and truncated to a system of ODEs (Equations (3.18)–(3.24)) through the Local Non-Similarity technique. This system is solved through a composite loss function, which includes momentum loss, energy loss, and boundary condition loss, each being the residual of the respective governing equation or condition. The solution is approximated by a PINN, and the performance of the PINN is assessed by comparing the solutions with a reference solution that has been achieved through a standard numerical procedure. The convergence and accuracy of the PINN model in simulating the flow and thermal behavior within the given physical constraints are verified by exploring the total loss. We define separate loss functions for each of the problem's boundary conditions. Each of the three major components of the overall loss function is data loss, physics-informed loss, and boundary condition loss is constructed to ensure that the solution of the neural network complies with the respective boundary constraints and differential equations. The following definitions hold for these components:

For physics loss we first define the residual functions as

$$R_f = \left(1 + \frac{1}{\beta}\right) f''' + f f'' - (1 + \xi \Lambda) f'^2 - \left(M + \frac{1}{Da}\right) f' + \frac{\sin \xi}{\xi}(\theta) \quad (3.28)$$

$$R_\theta = \frac{1}{pr} \left[1 + \frac{4}{3} Rd \right] \theta'' + f \theta' \quad (3.29)$$

$$L_{physics} = \frac{1}{N} \sum_{i=1}^N (R_f^2 + R_\theta^2 + L_{BC}) \quad (3.30)$$

Boundary condition loss can be defined as:

$$L_{BC} = \frac{1}{N} \sum_{i=1}^N [f'(0)^2 + f(0)^2 + (\theta(0) - 1)^2 + f'(\infty)^2 + \theta(\infty)^2] \quad (3.31)$$

Data Loss is define as

$$L_{data} = \frac{1}{N} \sum_{i=1}^N (\hat{y}_i - y_i)^2 \quad (3.32)$$

Then the total loss can be calculated as:

$$Total\ Loss = L_{data} + L_{physics} + L_{BC} \quad (3.33)$$

3.5 Results and Discussion

To start the discussion, we first investigate the PDEs Eqn. (3.6)-(3.8) which is computed by a well-known Local Non-Similarity method. Applying first level of truncation of Local Non-Similarity methods to Eqn. (3.6)-(3.8), the given PDEs is truncated to system of ODEs given in Eqn. (3.18)-(3.24). The system of ODEs then solved using PINN. Solution process is started by developing a neural network with thirty-two number of neuron and four number of hidden layers. For Adam optimizer the learning rate is set as 10^{-4} . The concern equation is also computes using a bvp4c method. The L^2 error is measure to examine the consistency of the predicted results. It is evaluated that the PINN result depicts excellent agreement with numerical solution, illustrating the consistency and reliability of our PINN methodology. However, it is noted that as the number of iteration and learning rate increases the L^2 error decreases and PINN predicted solution and numerically computed solution match very well. The computed PINN solution and Numerical solution of f, f', f'' and θ, θ' are depicted in Fig. 3.3 using one thousand number of iterations. It is observed that the PINN predicted solution and the Numerical solution are far from each other. To achieve best results so that PINN and Numerical solution matched well we increased the number of iteration up to

twenty-five thousands as well as increase the number of hidden layer and neuron and achieve well matched solution (PINN and Numerical) that is shown in Fig. 3.4. Fig. 3.5 shows the total loss function and physics loss of the computed solution. L^2 Error describes about the difference between NN predicted results and numerical results. L^2 Error of our consider problem for various number of iteration and learning rate are illustrated in Table 3.1

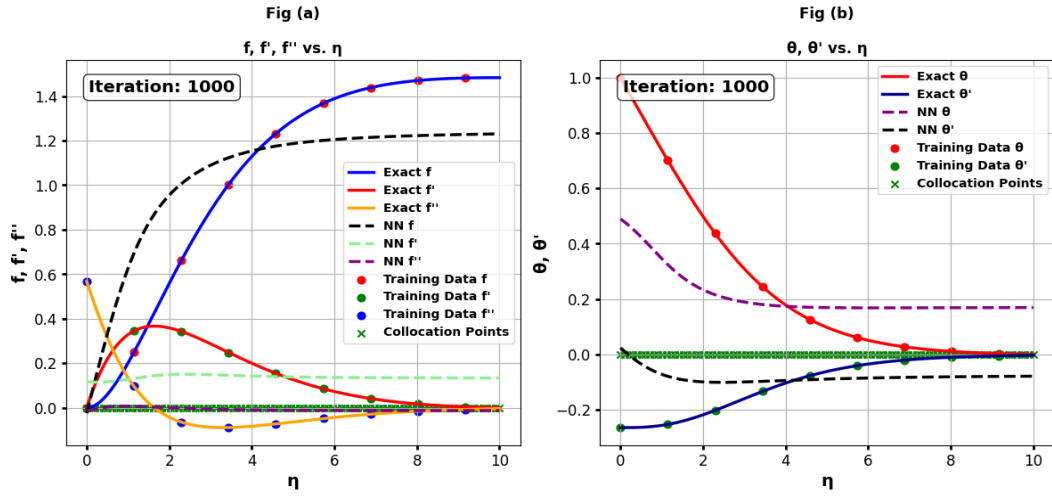


Fig. 3.2: PINN vs. numerical solution using 1000 epochs.

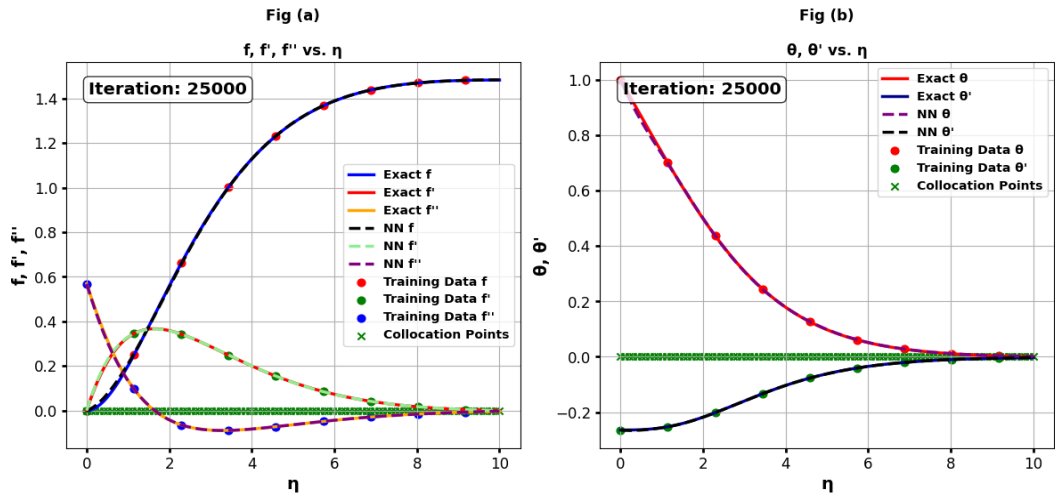


Fig. 3.3: PINN vs. numerical solution using 25000 epochs and deeper network.

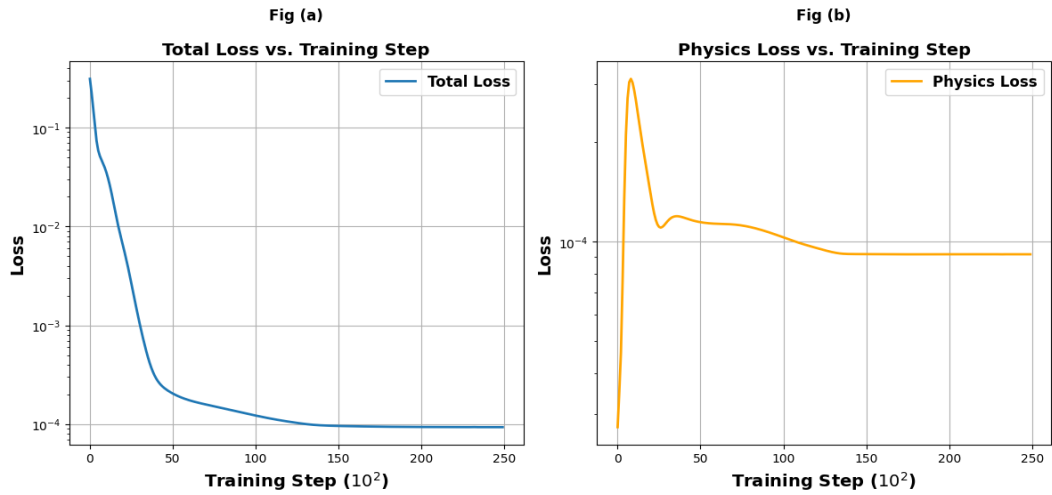


Fig. 3.4: Total and physics-informed loss convergence during training.

Table 3. 2: L^2 error between the predicted and numerical solution for different number of iterations and training steps.

Index	Iteration	Training Step	L^2 Error	Index	Iteration	Training Step	L^2 Error
0	100	10000	0.560829	119	12000	1200000	0.004366
9	1000	100000	0.174006	129	13000	1300000	0.004468
19	2000	200000	0.073053	139	14000	1400000	0.004526
29	3000	300000	0.029032	149	15000	1500000	0.004196
39	4000	400000	0.016682	159	16000	1600000	0.003862
49	5000	500000	0.014043	169	17000	1700000	0.003606
59	6000	600000	0.012162	179	18000	1800000	0.003418
69	7000	700000	0.010637	189	19000	1900000	0.003284
79	8000	800000	0.009175	199	20000	2000000	0.003184
89	9000	900000	0.007606	219	22000	2200000	0.003018
99	10000	1000000	0.006101	239	24000	2400000	0.002909
109	11000	1100000	0.004943				

The tables 3.3-3.6 show the effect of hidden layers, neurons, and mesh size (δ) on the L^2 error of velocity f , temperature θ , and their derivatives. Tables 3.3 and 3.4 analyze a fixed domain size $\eta=5$ and then Tables 3.5 and 3.6 take the computational domain out to $\eta = 10$. In each situation, adding more hidden layers and neurons results in a considerable decrease in L^2 error, which is an evidence to the relevance of network depth and width for enhancing model accuracy. Comparing mesh sizes, a smaller mesh size ($\delta=0.02$) always gives smaller errors than a larger mesh size ($\delta = 0.05$), as evident from comparing Table 3.3 vs. Table 3.4 and Table 3.5 vs. Table 3.6. For instance, the L^2 error for f goes from 0.1489 to 0.0689 in Table 3.3 and from 0.1472 to 0.0406 in Table 3.4 as the mesh size is made smaller.

However, a significant observation is made when considering the impact of expanding the number of neurons (network width) with a constant number of hidden layers. Table 3.4 illustrates that merely expanding the number of neurons from 8 to 64 at $\eta = 5$ fails to notably improve solution accuracy; although the loss function does reduce, improvement in L^2 error is negligible. This implies that the issue is not one of the

network topology but of the small domain size. To avoid this, the third approach enlarges the domain size to $\eta = 10$ with a fine mesh size $\delta = 0.02$, as observed in Table 3.6. This method, coupled with the added number of neurons, enhances the accuracy of the solution. As compared to Table 3.4, Table 3.6 has significantly reduced L^2 errors with the same number of neurons, verifying that the combination of a large enough computational domain and broader network architecture provides better performance. For example, the L^2 error for f is reduced to 0.0112 in Table 3.6 from 0.0406 in Table 3.4 despite having the same number of layers and neurons. These results verify that the robustness and accuracy of PINN solutions are extremely sensitive to both the selection of domain size and network architecture, and are best acquired through the accurate adjustment of these hyper-parameters.

Table 3. 3: L^2 errors of f, f', f'', θ and θ' using PINN for $\eta = 5$ and finer mesh size $\delta = 0.05$ with varying hidden layers and neurons

Index	Hidden layers	Neurons	L^2 error of $f(\eta)$	L^2 error of $f'(\eta)$	L^2 error of $f''(\eta)$	L^2 error of $\theta(\eta)$	L^2 error of $\theta'(\eta)$
0	4	8	0.148939	0.142604	0.043511	0.075944	0.033358
1	4	16	0.068957	0.064673	0.044154	0.042902	0.048436
2	4	32	0.040901	0.030414	0.031113	0.023889	0.003824
3	4	64	0.017434	0.015053	0.022886	0.039255	0.016300

Table 3. 4: L^2 errors of f, f', f'', θ and θ' using PINN for $\eta = 5$ and finer mesh size $\delta = 0.02$ with varying hidden layers and neurons

Index	Hidden layers	Neurons	L^2 error of $f(\eta)$	L^2 error of $f'(\eta)$	L^2 error of $f''(\eta)$	L^2 error of $\theta(\eta)$	L^2 error of $\theta'(\eta)$
0	4	8	0.147182	0.136272	0.070664	0.103261	0.033256
1	4	16	0.040688	0.059677	0.022326	0.029227	0.053189
2	4	32	0.019299	0.014726	0.013675	0.011647	0.004402
3	4	64	0.014152	0.012186	0.012448	0.006135	0.003267

Table 3. 5: L^2 errors of f, f', f'', θ and θ' using PINN for $\eta = 10$ and finer mesh size $\delta = 0.05$ with varying hidden layers and neurons

Index	Hidden layers	Neurons	L^2 error of $f(\eta)$	L^2 error of $f'(\eta)$	L^2 error of $f''(\eta)$	L^2 error of $\theta(\eta)$	L^2 error of $\theta'(\eta)$
0	4	8	0.268813	0.163087	0.057719	0.113374	0.050122
1	4	16	0.047539	0.058031	0.057354	0.050636	0.041342
2	4	32	0.032227	0.026190	0.028598	0.020872	0.004863
3	4	64	0.017699	0.018704	0.022899	0.021682	0.006868

Table 3. 6: L^2 errors of f, f', f'', θ and θ' using PINN for $\eta = 10$ and finer mesh size $\delta = 0.02$ with varying hidden layers and neurons

Index	Hidden layers	Neurons	L^2 error of $f(\eta)$	L^2 error of $f'(\eta)$	L^2 error of $f''(\eta)$	L^2 error of $\theta(\eta)$	L^2 error of $\theta'(\eta)$
0	4	8	0.253188	0.159095	0.075647	0.145722	0.044483
1	4	16	0.038113	0.056815	0.059722	0.044117	0.034429
2	4	32	0.019958	0.015051	0.015451	0.013998	0.004193
3	4	64	0.011155	0.008880	0.007168	0.006232	0.002676

The next step is to evaluate the comparison between numerical results and PINNs predicted results for varying parameter values. For this purpose, we take one parameter values varying while the rest of parameter values are keeping constant. Then we train our neural network for different values of parameter and finally by taking one hundred and fifty-six number of neurons, eight number of hidden layer and learning rate $\lambda = 0.0001$, over numerical results and PINNs prediction are match and ensuing very small L^2 error.

The effect of Prandtl number Pr on velocity and temperature profile is examined. Higher Prandtl number implies that momentum transfer is more capable than heat transfer. This result to a thinner velocity boundary layer near the cylinder, leading to reduction of velocity profile. So, as the Pr increase the velocity of fluid become decrease. Conversely, higher Prandtl number leads to thicker thermal boundary layer, implying that temperature gradient near the cylinder surface is smaller resulting in a high temperature closer to the cylinder. Fig. 3.6 demonstrates the velocity and temperature profile for different values of Pr . The L^2 error for varying number of Pr are given in Table 3.7. The PINNs predicted results are represented by dotted lines and numerical results are shown by solid lines. It is observed that both results are matched.

Table 3. 7: L^2 -error of $f'(\eta)$ and $\theta(\eta)$ for varying values of Pr .

index	Pr	L^2 error of $f'(\eta)$	L^2 error of $\theta(\eta)$
0	1.7	0.005598	0.001599
1	5.7	0.020155	0.003462
2	7.7	0.005921	0.001649
3	9.7	0.005051	0.003471

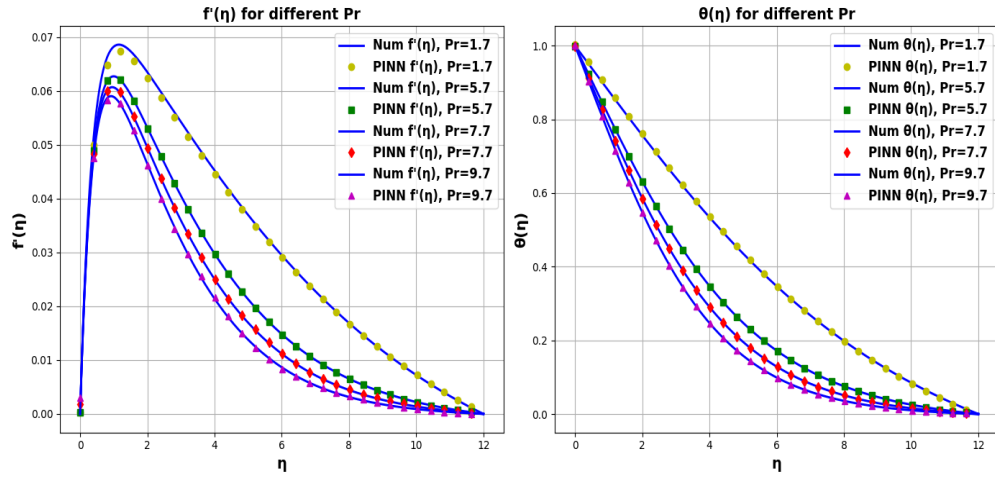


Fig. 3.5: Velocity and temperature profiles for different values Pr with PINN and numerical solutions.

Effect of magnetic effect on fluid velocity and temperature is examined in the current study. For different values of magnetic parameter M both the numerical results and PINNs result are plotted in Fig. 3.7. It is depicted that both PINNs and Numerical results are overlapping which shows better accuracy of the predicted PINNs solution. The L^2 error for varying values of magnetic parameter M is represented in Table 3.8. Furthermore, it is detected that as the values of M enhances the Lorentz force become stronger leading to increase the resistance to the flow thus lowering the velocity of the fluid. As magnetic effect M increase the temperature rises because of additional energy lead by Lorentz forces.

Table 3. 8: L^2 -error of $f'(\eta)$ and $\theta(\eta)$ for varying values of M .

index	M	L^2 error of $f'(\eta)$	L^2 error of $\theta(\eta)$
0	1.5	0.007729	0.001469
1	3.5	0.003324	0.000793
2	6.5	0.011567	0.000924
3	7.5	0.004145	0.000515

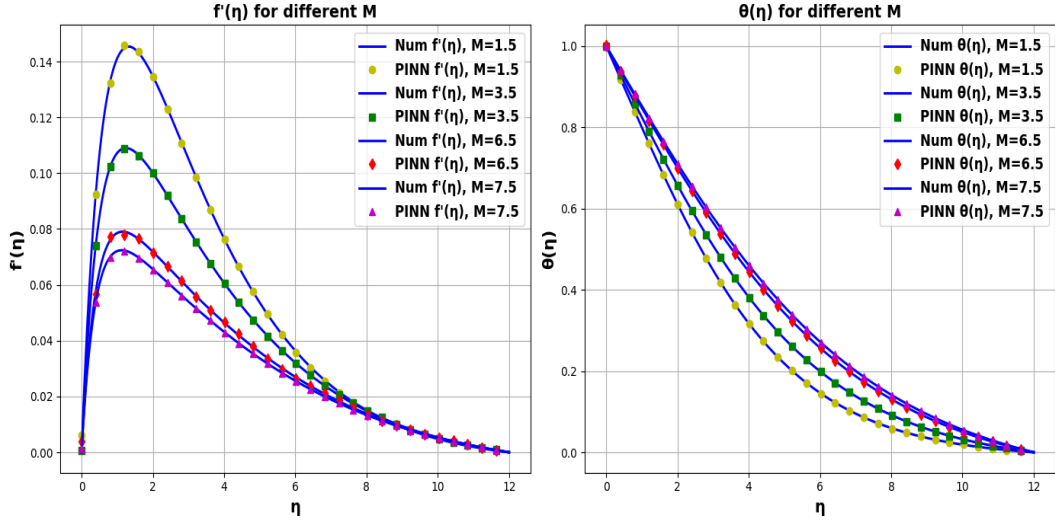


Fig. 3.6: Velocity and temperature profiles for different values M with PINN and numerical solutions.

The Darcy parameter Da has linear relationship with permeability of the porous medium. As Da increases it shows that the medium is more permeable which lead to low resistance to the fluid flow and hence velocity of the fluid rises. A higher Darcy number cause lower temperature because the fluid flow through more permeable porous medium face less resistance which reduces the heat generation . So the temperature of the fluid decrease as Darcy umber Da increases. Fig. 3.8 illustrate the effect of Da on velocity and temperature, both numerical and PINNs results are depicted for various values of Da . Table 3.9 symbolizes the L^2 error between the PINNs and Numerical results for numerous values of Da .

Table 3. 9: L^2 -error of $f'(\eta)$ and $\theta(\eta)$ for varying values of Da .

Index	Da	L^2 error of $f'(\eta)$	L^2 error of $\theta(\eta)$
0	0.2	0.007851	0.001058
1	0.4	0.002301	0.000371
2	0.5	0.000909	0.00026
3	0.6	0.003406	0.001206

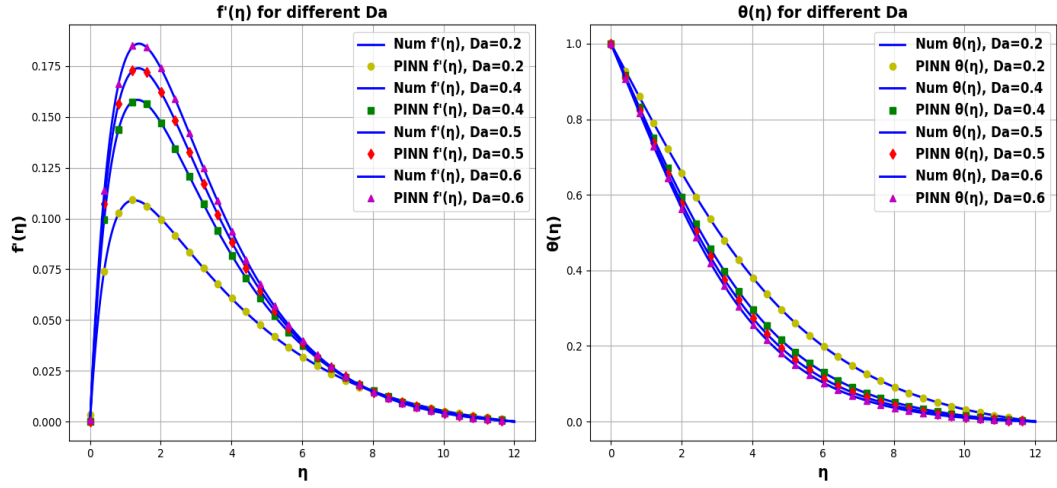


Fig. 3.7: Velocity and temperature profiles for different values Da with PINN and numerical solutions.

Effect of Radiation parameter on velocity and temperature is discussed in the current study. In Fig. 3.9 both the numerical and PINNs results are illustrated for various number of Radiation parameter Rd . Table 3.10 present the L^2 error between PINNs predicted results and Numerical results which show better consistency. Furthermore it is examine that when the radiation parameter is raised the fluid receive more thermal radiation which leads to raise the temperature of the fluid. As by increasing the Radiation parameter Rd , temperature enhances lead to reduce the viscosity of non-Newtonian fluid. This reduction in viscosity of the fluid facilitate fluid to move easily and hence the velocity of the fluid slightly increases.

Table 3. 10: L^2 -error of $f'(\eta)$ and $\theta(\eta)$ for varying values of Rd .

Index	Rd	L^2 error of $f'(\eta)$	L^2 error of $\theta(\eta)$
0	0.1	0.020539	0.000976
1	0.3	0.003182	0.000556
2	0.6	0.003282	0.003078
3	0.8	0.003133	0.000412

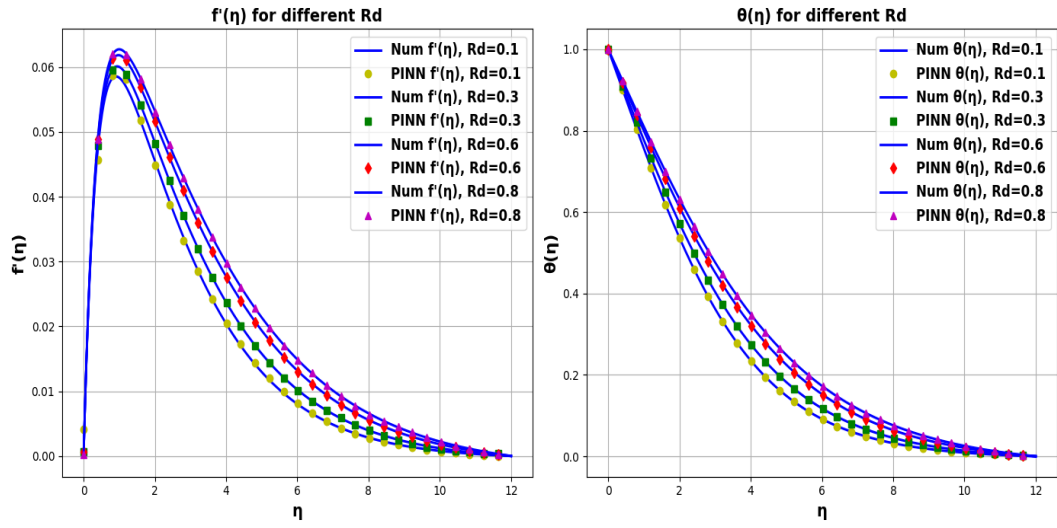


Fig. 3.8: Velocity and temperature profiles for different values Rd with PINN and numerical solutions.

3.4 Conclusion

The current research uses a Physics-Informed Neural Network (PINN) scheme to study the boundary layer flow and heat transfer behavior of non-Newtonian fluid flow around an embedded horizontal cylinder placed in a non-Darcy porous medium under a transverse magnetic field. The Local Non-Similarity technique is utilized to transform the nonlinear partial differential governing equations into a system of ordinary differential equations, and the resulting equations are solved with PINNs in PyTorch (Python). The study highlights the importance of selecting optimal neural network hyper parameters such as learning rate, number of hidden layers, neurons per layer, and training iterations for achieving reliable and precise results. A learning rate of less than 10^{-4} was shown to produce very accurate predictions. As illustrated in Fig. 3.1 and Fig. 3.2, the comparison of PINN outcomes with traditional numerical solutions shows a high degree of agreement, confirming the efficacy of the suggested method. A detailed parametric analysis is conducted to examine the impact of various physical parameters on the velocity and temperature profiles. The results indicate the following:

When the Prandtl number Pr rises, the temperature and velocity profiles simultaneously decrease, suggesting less thermal diffusion.

An increase in the magnetic parameter M suppresses velocity due to the Lorentz force, while slightly elevating the temperature due to resistive heating.

Higher Darcy number Da enhances velocity by increasing permeability, while slightly decreasing the temperature field due to enhanced fluid infiltration.

An increase in the radiation parameter Rd results in elevated velocity and temperature profiles due to enhanced radiative heat transfer effects within the boundary layer.

From an industrial standpoint, the study's findings are useful for processes involving flow control and thermal management in porous media environments, such as heat exchangers, thermal insulation systems, energy storage devices, and magnetic field-assisted flow systems in the polymer processing and biomedical industries.

On the computational front, this research demonstrates the capability of PINNs as a robust and flexible alternative to traditional numerical solvers.

References

- [1] Subba Rao, A., Ramachandra Prasad, V., Bhaskar Reddy, N., & Anwar Bég, O. (2015). Heat transfer in a Casson rheological fluid from a semi-infinite vertical plate with partial slip. *Heat Transfer—Asian Research*, 44(3), 272-291.
- [2] Ramachandra Prasad, V., Abdul Gaffar, S., & Rushi Kumar, B. (2019). Non-similar computational solutions for double-diffusive MHD transport phenomena for non-newtnian nanofluid from a horizontal circular cylinder. *Nonlinear Engineering*, 8(1), 470-485.
- [3] Gaffar, S. A., Prasad, V. R., & Reddy, E. K. (2017). Magnetohydrodynamic free convection flow and heat transfer of non-Newtonian tangent hyperbolic fluid from horizontal circular cylinder with Biot number effects. *International Journal of Applied and Computational Mathematics*, 3(2), 721-743.
- [4] Bin Turabi, Y. U. U., Munir, S., & Amin, A. (2023). Numerical analysis of convective transport mechanisms in two-layer ternary ($\text{TiO}_2\text{-SiO}_2\text{-Al}_2\text{O}_3$) Casson hybrid nanofluid flow in a vertical channel with heat generation effects. *NUMERICAL HEAT TRANSFER PART A-APPLICATIONS*.
- [5] Amin, A., Munir, S., & Farooq, U. (2024). Flow dynamics and convective transport analysis of two-layered dissipative Casson hybrid nanofluid flow in a vertical channel. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 238(2), 395-404.
- [6] Das, S., Mahato, N., Ali, A., & Jana, R. N. (2022). Dynamical behaviour of magneto-copper-titania/water-ethylene glycol stream inside a gyrating channel. *Chemical Physics Letters*, 793, 139476.
- [7] Ali, A., Das, S., & Jana, R. N. (2023). MHD gyrating stream of non-Newtonian modified hybrid nanofluid past a vertical plate with ramped motion, Newtonian

- heating and Hall currents. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, 103(9), e202200080.
- [8] Hussain, S., Islam, S., Raja, M. A. Z., Nisar, K. S., & Shoaib, M. (2024). Intelligent computing technique to study heat and mass transport of Casson nanofluidic flow model on a nonlinear slanted extending sheet. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, 104(3), e202300224.
- [9] Mishra, V. K., & Chaudhuri, S. (2021). Genetic algorithm-assisted artificial neural network for retrieval of a parameter in a third-grade fluid flow through two parallel and heated plates. *Heat Transfer*, 50(3), 2090-2128.
- [10] Yadav, D., Naruka, D. S., & Singh, P. K. (2021). The insight flow characteristics of concentrated MWCNT in water base fluid: experimental study and ANN modeling. *Heat and Mass Transfer*, 57(11), 1829-1844.
- [11] Awais, M., Rehman, H., Raja, M. A. Z., Awan, S. E., Ali, A., Shoaib, M., & Malik, M. Y. (2022). Hall effect on MHD Jeffrey fluid flow with Cattaneo–Christov heat flux model: An application of stochastic neural computing. *Complex & Intelligent Systems*, 8(6), 5177-5201.
- [12] Tian, J., Qi, C., Sun, Y., Yaseen, Z. M., & Pham, B. T. (2021). Permeability prediction of porous media using a combination of computational fluid dynamics and hybrid machine learning methods. *Engineering with Computers*, 37, 3455-3471.
- [13] Tizakast, Y., Kaddiri, M., Lamsaadi, M., & Makayssi, T. (2023). Machine Learning based algorithms for modeling natural convection fluid flow and heat

- and mass transfer in rectangular cavities filled with non-Newtonian fluids. *Engineering Applications of Artificial Intelligence*, 119, 105750.
- [14] Sofos, F., & Karakasidis, T. E. (2021). Machine Learning techniques for fluid flows at the nanoscale. *Fluids*, 6(3), 96.
 - [15] Wang, Q., Yan, L., Hu, G., Li, C., Xiao, Y., Xiong, H., & Noack, B. R. (2022). DRLinFluids: An open-source Python platform of coupling deep reinforcement learning and OpenFOAM. *Physics of Fluids*, 34(8).
 - [16] Ballinas, M. R., Bedle, H., & Devegowda, D. (2023). Supervised machine learning for discriminating fluid saturation and presence in subsurface reservoirs. *Journal of Applied Geophysics*, 217, 105192.
 - [17] Srilatha, P., Gowda, R. P., Madhu, J., Nagaraja, K. V., Gamaoun, F., Kumar, R. V., & Karthik, K. (2024). Designing a solid–fluid interface layer and artificial neural network in a nanofluid flow due to rotating rough and porous disk. *Journal of Thermal Analysis and Calorimetry*, 149(2), 867-878.
 - [18] Cuomo, S., Di Cola, V. S., Giampaolo, F., Rozza, G., Raissi, M., & Piccialli, F.(2022). Scientific machine learning through physics–informed neural networks: Where we are and what’s next. *Journal of Scientific Computing*, 92(3), 88.
 - [19] Baty, H., & Baty, L. (2023). Solving differential equations using physics informed deep learning: a hand-on tutorial with benchmark tests. *arXiv preprint arXiv:2302.12260*.
 - [20] Almqvist, A. (2021). Fundamentals ofg physics-informed neural networks applied to solve the Reynolds boundary value problem. *Lubricants*, 9(8), 82.

- [21] Kharazmi, E., Zhang, Z., & Karniadakis, G. E. (2019). Variational physics-informed neural networks for solving partial differential equations. *arXiv preprint arXiv:1912.00873*.
- [22] Fallah, A., & Aghdam, M. M. (2024). Physics-Informed Neural Network for Solution of Nonlinear Differential Equations. In *Nonlinear Approaches in Engineering Application: Automotive Engineering Problems* (pp. 163-178). Cham: Springer Nature Switzerland.
- [23] Cai, S., Wang, Z., Wang, S., Perdikaris, P., & Karniadakis, G. E. (2021). Physics informed neural networks for heat transfer problems. *Journal of Heat Transfer*, 143(6), 060801.
- [24] Sharma, P., Chung, W. T., Akoush, B., & Ihme, M. (2023). A review of physics-informed machine learning in fluid mechanics. *Energies*, 16(5), 2343.
- [25] Dieva, N., Aminev, D., Kravchenko, M., & Smirnov, N. (2024). Overview of the application of physically informed neural networks to the problems of nonlinear fluid flow in porous media. *Computation*, 12(4), 69.
- [26] Eivazi, H., Wang, Y., & Vinuesa, R. (2024). Physics-informed deep-learning applications to experimental fluid mechanics. *Measurement science and technology*, 35(7), 075303.
- [27] Nguyen, T. N. K., Dairay, T., Meunier, R., & Mougeot, M. (2022). Physics-informed neural networks for non-Newtonian fluid thermo-mechanical problems: An application to rubber calendering process. *Engineering Applications of Artificial Intelligence*, 114, 105176.
- [28] Raissi, M., Perdikaris, P., & Karniadakis, G. E. (2017). Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations. *arXiv preprint arXiv:1711.10561*.

- [29] Raissi, M., Perdikaris, P., & Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378, 686-707.
- [30] Bararnia, H., & Esmaeilpour, M. (2022). On the application of physics informed neural networks (PINN) to solve boundary layer thermal-fluid problems. *International Communications in Heat and Mass Transfer*, 132, 105890.
- [31] Ramachandra Prasad, V., Subba Rao, A., & Anwar Bég, O. (2013). Flow and heat transfer of Casson fluid from a horizontal circular cylinder with partial slip in non-Darcy porous medium. *J Appl Computat Math*, 2(127), 2.
- [32] Waqas, H., Naqvi, S. M. R. S., Alqarni, M. S., & Muhammad, T. (2021). Thermal transport in magnetized flow of hybrid nanofluids over a vertical stretching cylinder. *Case Studies in Thermal Engineering*, 27, 101219.
- [33] Minkowycz, W. J., Sparrow, E. M., Schneider, G. E., & Pletcher, R. H. (1988). *Handbook of numerical heat transfer*.
- [34] Minkowycz, W. J., & Sparrow, E. M. (1978). Numerical solution scheme for local nonsimilarity boundary-layer analysis. *Numerical Heat Transfer, Part B: Fundamentals*, 1(1), 69-85.
- [35] Laabid, Z., Moumen, A., Mansouri, K., & Siadat, A. (2023). Numerical study of the speed's response of the various intelligent models using the tansig, logsig and purelin activation functions in different layers of artificial neural network. *IAES International Journal of Artificial Intelligence*, 12(1), 155.

