Weekly Report

# Physics Informed Neural Networks

Supervisor - Prof. Kano Sensei

By - Aman Khilani

# Contents

# Introduction

A Continuous Stirred Tank Reactor (CSTR) is a common type of chemical reactor which is typically used for liquid-phase reactions. It is a very basic and simple model.

- A CSTR **operates continuously**, with reactants being continuously fed into the reactor and products being continuously removed. This ensures that the reactor operates at a steady state.
- The contents are continuously stirred with an agitator, ensuring **uniform composition and temperature** throughout the reactor.
- In the given problem, the chemical reaction occurring within the CSTR is a **first-order reaction** with respect to the concentration of the reactant. The reaction rate depends on the concentration of the reactant and follows the **Arrhenius temperature dependence**, meaning the reaction rate increases exponentially with temperature.

**Problem Statement -** As of now, I've used the same problem statement as given in the Appendix C of the book **Process Control**, by Sir Thomas E. Marlin.
The objective is to raise the temperature of a chemical reactor to 395.3 K without exceeding this temperature by adjusting the coolant flow.

# Introduction

The system is liquid in the tank, and the **reaction parameters** are as follows:

- Flow Rate (F): 1 $m^3$/min
- Volume (V): 1 $m^3$
- Initial Concentration of Reactant ($CA_o$): 2.0 kmol/$m^3$
- Initial Temperature ($T_o$): 323 K
- Specific Heat Capacity of Liquid (Cp): 1 cal/(g·K)
- Density of Liquid (ρ): 106 g/$m^3$
- Reaction Rate Constant ($k_o$): 1.0 × $10^{10}$ $min^{-1}$
- Activation Energy divided by Gas Constant (E/R): 8330.1 K
- Heat of Reaction (ΔH_rxn): -130 × $10^6$ cal/kmol
- Coolant Inlet Temperature (Tcin): 365 K
- Steady-State Coolant Flow Rate (Fc)□: 15 $m^3$/min
- Specific Heat Capacity of Coolant (Cp_c): 1 cal/(g·K)
- Density of Coolant (ρ_c): 106 g/$m^3$
- Heat Transfer Coefficient (a): 1.678 × $10^6$ (cal/min)/(K)
- Coolant Flow Exponent (b): 0.5

The steady-state values of the dependent variables are given as:

- Steady-State Temperature (T□): 394 K
- Steady-State Concentration of Reactant (CA□): 0.265 kmol/$m^3$

A step change in the coolant flow of -1 $m^3$/min is applied to analyze its effect on the system.

# Introduction

## Governing Equations

- **Material Balance for the Reactant (A) :**

$$V\frac{dC_A}{dt} = F(C_{A0} - C_A) - Vk_0e^{-\frac{E}{RT}}C_A$$

Where:

- $C_A$ is the concentration of reactant A in the reactor.
- $F$ is the flow rate of the reactant.
- $V$ is the volume of the reactor.
- $C_{A0}$ is the initial concentration of the reactant.
- $k_0$ is the pre-exponential factor of the reaction rate.
- $E/R$ is the activation energy divided by the gas constant.
- $T$ is the temperature of the reactor.

- **Energy Balance :**

$$V\rho C_p\frac{dT}{dt} = \rho C_pF(T_0 - T) - \left(\frac{aFc^{b+1}}{Fc + aFc^b/2(\rho_c)C_{pc}}\right)(T - T_{cin}) + (-\Delta H_{rxn})Vk_0e^{-\frac{E}{RT}}C_A$$

Where:

- $T$ is the temperature of the reactor.
- $T_0$ is the initial temperature of the reactant.
- $T_{cin}$ is the inlet temperature of the coolant.
- $Fc$ is the flow rate of the coolant.
- $\rho$ and $C_p$ are the density and specific heat capacity of the liquid in the reactor.
- $\rho_c$ and $C_{pc}$ are the density and specific heat capacity of the coolant.
- $a$ is the heat transfer coefficient.
- $b$ is the coolant flow exponent.
- $\Delta H_{rxn}$ is the heat of reaction.

# Simulation & Data

1. Simulation Setup: The provided CSTR simulation was set up to run for 10 Sec with a step change in coolant flow applied at t = 0. Time step for the simulation was chosen to be sufficiently small to capture the dynamics of the system accurately.
2. Step Change: A step change of -1 $m^3$/min in the coolant flow rate was introduced to observe its effect on the reactor temperature and concentration of reactants.
3. Based on this simulation, a generated a CSV file which was later used in training and testing my PINN model.
4. **Data-analysis**: Based on the data generated, it was seen that as the coolant temperature was reduced over a period of time, the reactor temperature also reduced (which was also theoretically correct).

# PINN Model

A PINN model basically helps in utilizing the underlying physical laws represented by differential equations to learn and predict system behaviors.

**Input Variables:**

- F2: Flow rate of the reactor ($m^3$/min)
- T1: Inlet temperature of the reactor (K)
- CA1: Initial concentration of species A (kmol/$m^3$)
- F1: Flow rate ($m^3$/min)
- Fc1: Coolant flow rate ($m^3$/min)
- Tc1: Coolant inlet temperature (K)
- Time: Time (min)

**Output Variables:**

- V: Volume of the reactor ($m^3$)
- CA: Concentration of species A (kmol/$m^3$)
- T: Temperature of the reactor (K)
- Tc_out: Outlet temperature of the coolant (K)

**Model Structure:**

- **Input Layer:** The input layer consists of 7 nodes corresponding to the input variables.
- **Hidden Layers:** The model includes 4 hidden layers with 512 hidden nodes and activation function - Tanh.
- **Output Layer:** The output layer consists of 4 nodes corresponding to the output variables.

# PINN Model

**Loss Function -** The loss function in a PINN model is a combination of the data loss and the physics-informed loss. The data loss used in the model is the Mean Squared Error (MSE) between the predicted and true values. The physics-informed loss is based on the differential equations governing the system.

**Training Process :**

- **Data Preparation:**
  1. Split the data into training and test sets based on ***test_size* = 0.2, *random_state* = 45**
  2. **Standardized** the input and output variables and converted them into **tensors**
  3. Created the training dataset and ***train_loader***
- **Model Initialization:**
  1. Created the reactor equation functions
  2. Defined the PINN model with the forward and loss functions
  3. Initialized the model, loss function and optimizer
- **Training Loop:** (for each epoch)
  1. Total ***100 epochs*** with ***batch_size* = 32**
  2. Forward propagation to compute predictions
  3. Calculating the combined loss (data loss + physics-informed loss)
  4. Backpropagation to update the model parameters

# PINN Model

- **Early Stopping:**
    1. After testing the initial model, there were some error due to overfitting
    2. The MSE for training was very high as compared to that for testing
    3. So I implemented some early stopping techniques like L2 Regularization and Dropout
    4. Then based on the best loss generated, a model was selected and then the evaluation was done based on that ***best_model.pth***

**Evaluation :**

- Based on the best model selected, there were 2 types of errors generated-
    1. MSE
    2. R2_Score

**Reference Paper used:**

1. As suggested by my supervisor, I found and read a paper about the implementation of PINN models on different chemical processes
2. The paper I referenced is - "**Optimal temperature trajectory for tubular reactor using physics informed neural networks**" by Rahul Patel, Sharad Bhartiya, Ravindra Gudi https://www.sciencedirect.com/science/article/pii/S0959152423000823#sec3.2
3. This paper implemented a similar approach in finding out the temperature trajectory for a tubular reactor based on the input variables using PINN
4. I also referenced the code they used and what losses they incorporated

# Results

## PINN2 (Basic ANN model)

- Train MSE: 0.1317, R2: 0.4294
- Test MSE: 0.0203, R2: 0.7264

## PINN3 (Improved model)

The loss function of the model was changed to add MSE loss

- Train MSE: 0.1078, R2: 0.5056
- Test MSE: 0.0160, R2: 0.7310

**Lowest MSE and Highest R2_Score**

## PINN3 (L2 Regularization)

L2 Regularisation and a learning rate scheduler are added.
Patience = 8, Patience_counter = 0

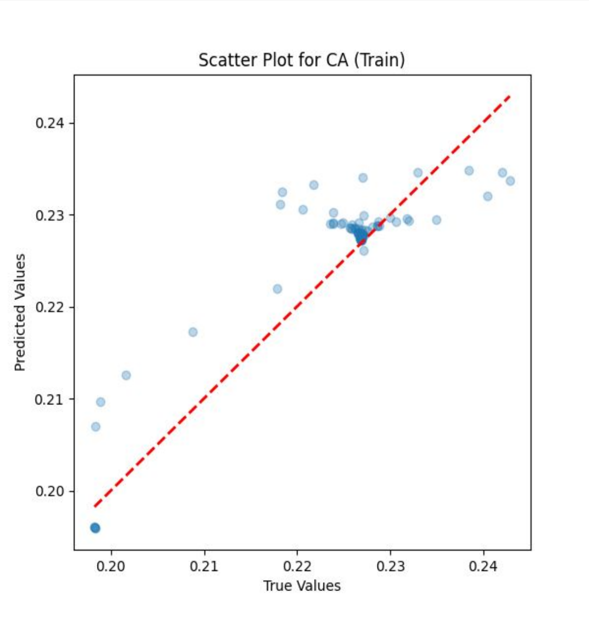
- Train MSE: 0.1316, R2: 0.4291
- Test MSE: 0.0252, R2: 0.7211

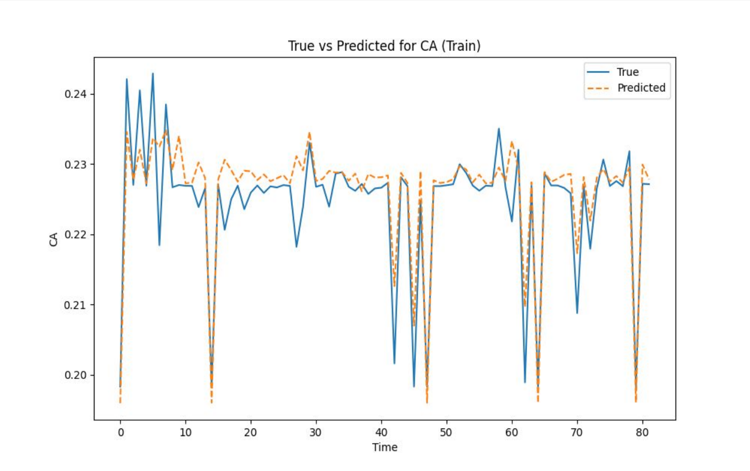
## PINN3 (Dropout for overfitting)

Added dropout to improve for overfitting. After hit and trial, the best patience level = 8.

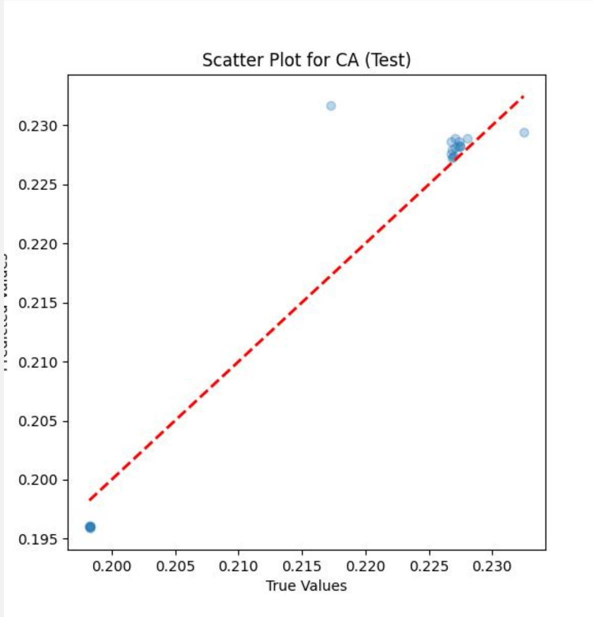
- Train MSE: 0.1357, R2: 0.4277
- Test MSE: 0.0348, R2: 0.7124

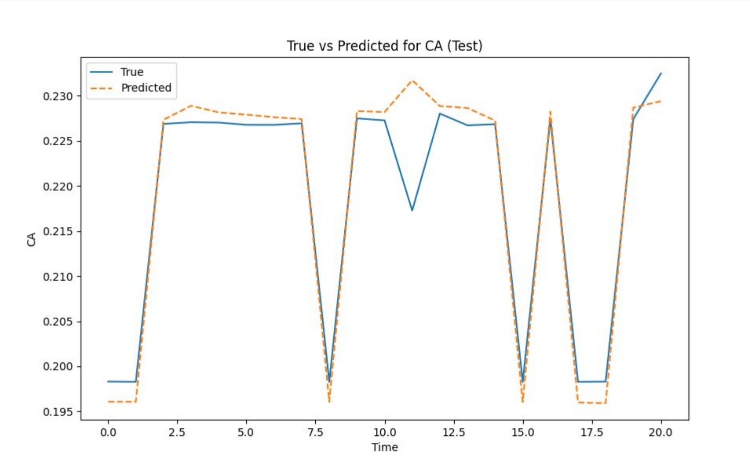
# Based on PINN3 (training) - Concentration (A)



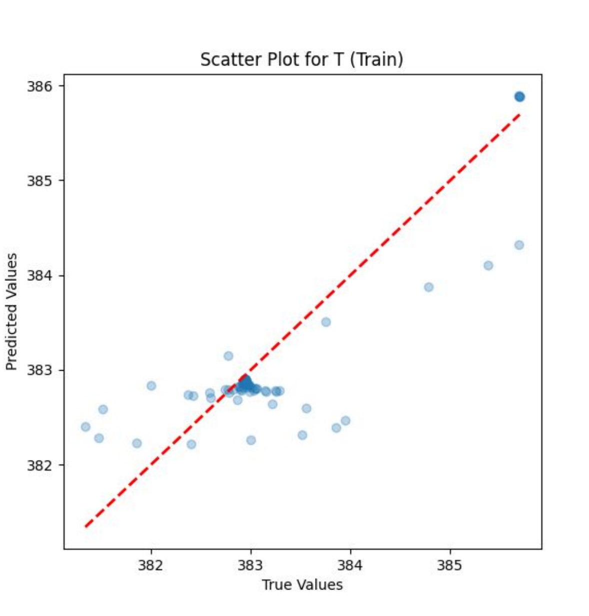



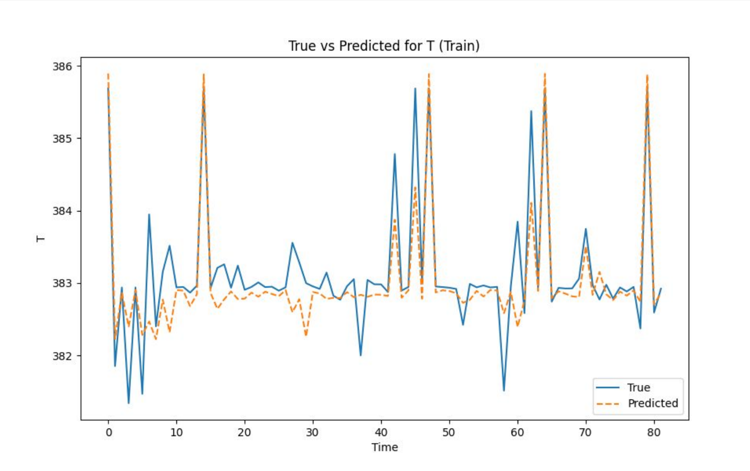
# Based on PINN3 (testing) - Concentration (A)



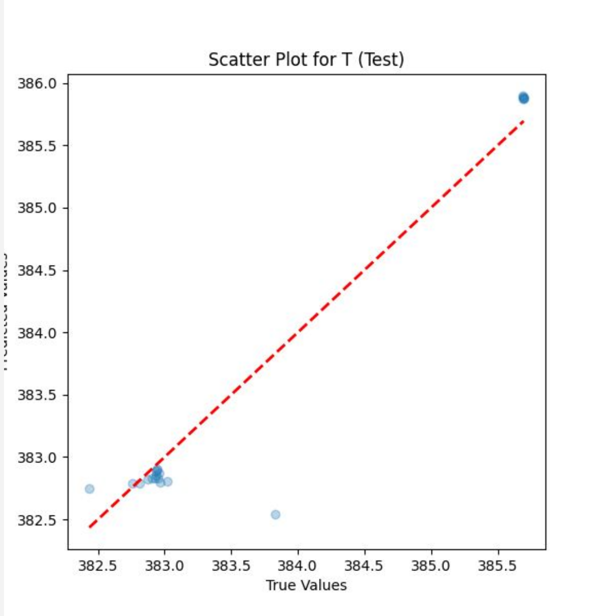



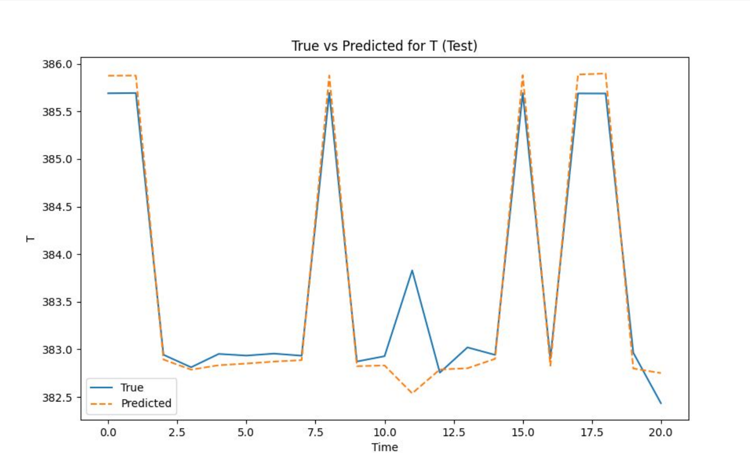
# Based on PINN3 (training) - Reactor Temperature



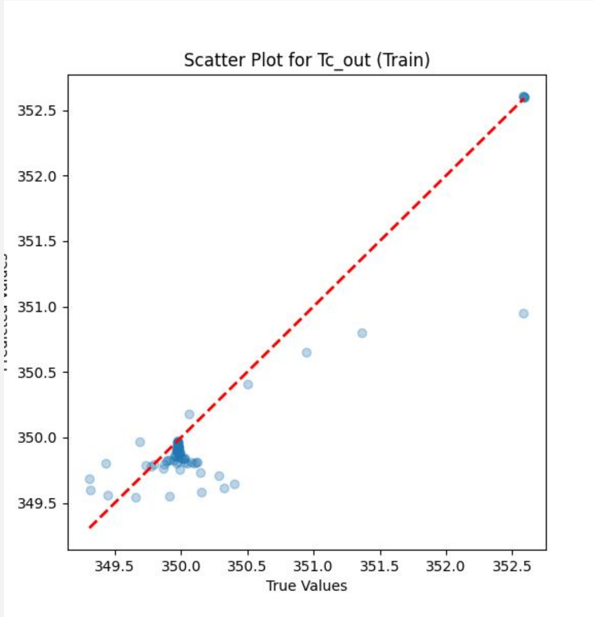



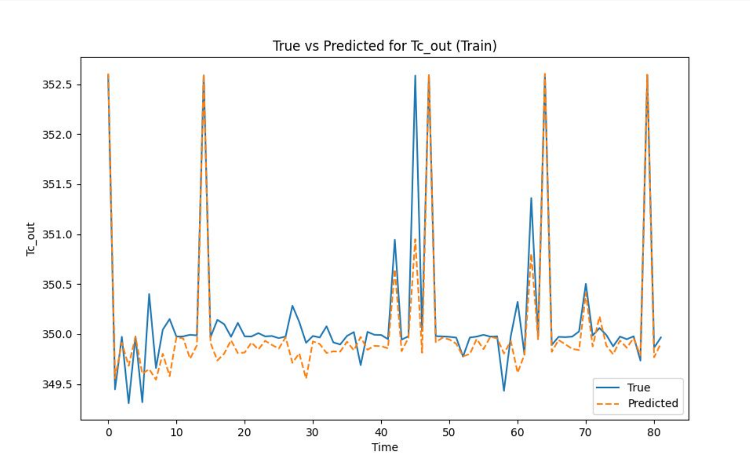
# Based on PINN3 (testing) - Reactor Temperature



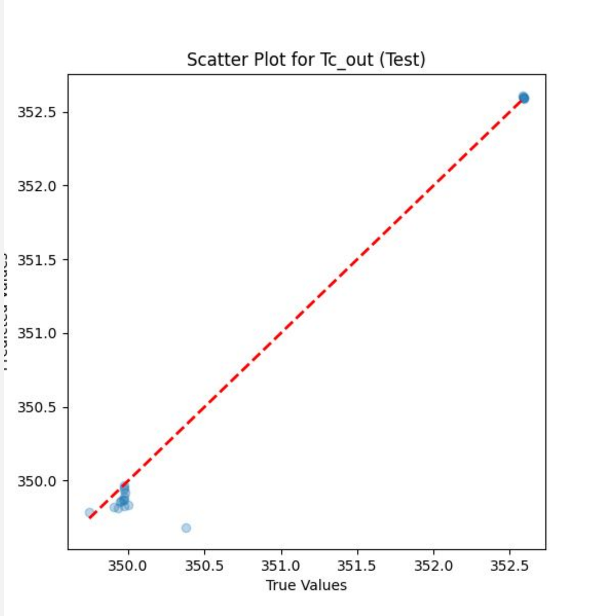



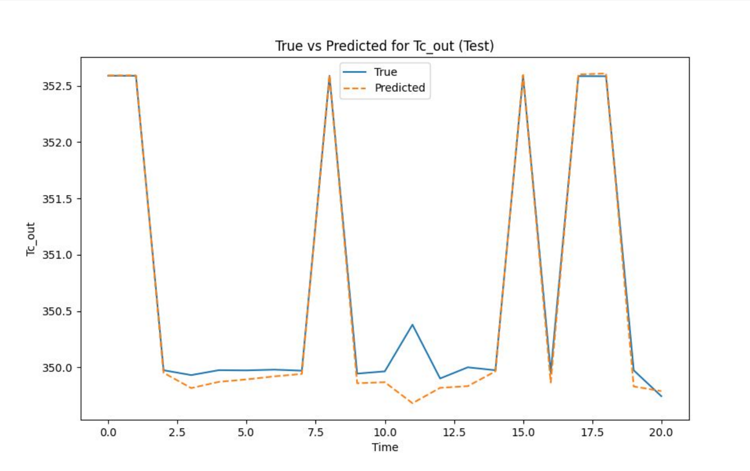
# Based on PINN3 (training) - Coolant Temperature



# Based on PINN3 (testing) - Coolant Temperature



# Conclusion

- Of all the variations, the PINN3 model has given the most promising results
- With an MSE as low as 0.016 and an R2 Score of as high as 73.10%
- I’ve also made a few changes in the code since last time
- The results could be further improved if we have more data