Non-invasive Measurement of Blood Flow Velocity, Viscosity and Density Using Physics-informed Neural Network

Akasapu Hemanthika
Ph.D. Student
School of Artificial Intelligence
Indian Institute of Technology, Delhi
aiz218616@scai.iitd.ac.in

Abstract

The mentioned work attempts to estimate blood flow velocity, viscosity and density at any point in the predetermined length of the artery. Trying to solve an inverse problem, a physics-informed neural network(PINN) trained on the sparse blood pressure data using a customised loss function is estimated to fulfil this task.

1 Introduction

With massive data and resources, present-day machine learning and deep learning models outperform most Artificial Intelligence (AI) tasks. Though model training uses a vast amount of data from different timestamps, there is still a possibility of missing prior system knowledge, which can be physics laws the system obeys. Due to this lack of integrating system physics into the trained model, one can never guarantee that the model converges in actuality. Adding prior system knowledge into the model can quickly move it in the right direction and generalize well with only a few training examples. This new area of AI which includes models built both as data and physics driven is popularly known as physics-informed deep learning or physics-informed machine learning.

Many researcher have and are continuously working on this area. Inspired with some of the researchers works an attempt is made to solve the inverse problem which is defined as: *given data and physics laws the system follows can we find out the model (approximation of the system) parameters?* For this task a neural network (physics-informed neural network PINN) is trained to find out the parameters of the model.

2 Proposed Methodology

The methodology chosen for accomplishing this task has four sub tasks: 1. Defining system physics - the physics laws the system is known to be followed, which are mostly in the form of partial differential equations; 2. Data preparation - data which will be used in model training, collocation points and boundary points (if any); 3. Modelling - finalizing neural network architecture, defining loss function; 4. Testing - testing trained model accuracy on new data. Figure 1 visualizes the flow of proposed methodology.

In the following sections all the four sub tasks of proposed methodology will be presented in detailed manner.

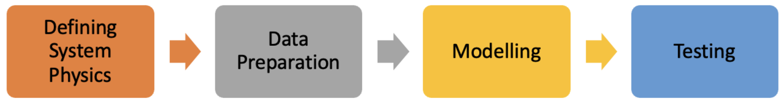

Figure 1: Proposed Methodology

2.1 Defining System Physics

The system selected in this work is an blood artery which is assumed to have blood flow in one direction. Figure 2 represents the pictorial representation of the system chosen.

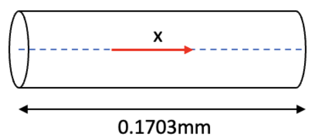

Figure 2: Assumed Artery

The following assumptions are made for the artery selected. 1. The blood is assumed to be Newtonian, in compressible fluid; 2. The cross-sectional area of the artery is not uniformed but has slight changes.

For defining the system physics, firstly inspired from the work of [1], Navier's stokes equation for one dimentional flow is considered which is given as below:

$$u_t + \lambda 1(uu_x) = p_x + \lambda 2(u_{xx}), x \in [0, 0.0494], t \in [0, 1] \tag{1}$$

where u(x,t) gives blood flow velocity, p(x,t) is pressure, $\lambda 1$ and $\lambda 2$ are blood density and viscosity, respectively.

In addition to the Navier Stokes equation, law of continuity and law of conservation of energy equations are also considered, which are defined as follows:

$$A_1 u_1 = A_2 u_2 \tag{2}$$

$$p_1 + \frac{1}{2}\rho u_1^2 = p_2 + \frac{1}{2}\rho u_2^2 \tag{3}$$

A is cross-sectional area, u blood velocity, p blood pressure.

2.2 Data Preparation

Data used for model training is extracted from paper [2]. For training only pressure data is used. We have pressure data of artery at point x=0.1, using this data and PDE conditions the model is trained. For testing data at point x=0 is used.

2.3 Modelling

The solution to above PDE can be obtained by modelling PINN which has two main sub networks: one network to predict blood velocity, cross-sectional area and pressure at given point and second to find the residue of the PDE.

The custom loss function used in PINN is designed as follows:

$$Loss = Loss_i + Loss_f \tag{4}$$

$$Loss_i = \frac{1}{N_i} \sum_{n=1}^{N_i} (p(x^i, t^i) - p^i)^2 \tag{5}$$

$$Loss_f = \frac{1}{N_f} \sum_{n=1}^{N_f} (f(x^i, t^i))^2 \tag{6}$$

$$f = u_t + \lambda 1(uu_x)p_x - \lambda 2(u_{xx}) \tag{7}$$

where $p(x^i, t^i)$ and p^i are predicted and labelled pressure, f is residual. N_i are the initial labelled training points and N_f are the randomly chosen collocation points in the domain. Therefore using customised loss function the model will be able to learn model parameters $\lambda 1$(1060 Kg/m^3), $\lambda 2$(3.5 mPa s) and unknown velocity u.

Below are the details of the parameters chosen for the best modelling iteration

- Optimizer - Adam
- Labelled data - 200 data points
- Collocation data - 5000 data points
- $\lambda 1$ and $\lambda 2$ initialization - 1060, 3.5
- Epochs - 20000 with learning rate 1e-3
- Model layers = [2, 100, 100, 100, 100, 100, 100, 100, 3]

3 Results

The best result obtained in the iteration when 20000 epochs and learning rate 1e-03 are chosen(Figure 3). Though the predicted values are not matching the exact values the model was able to capture non-linearity.

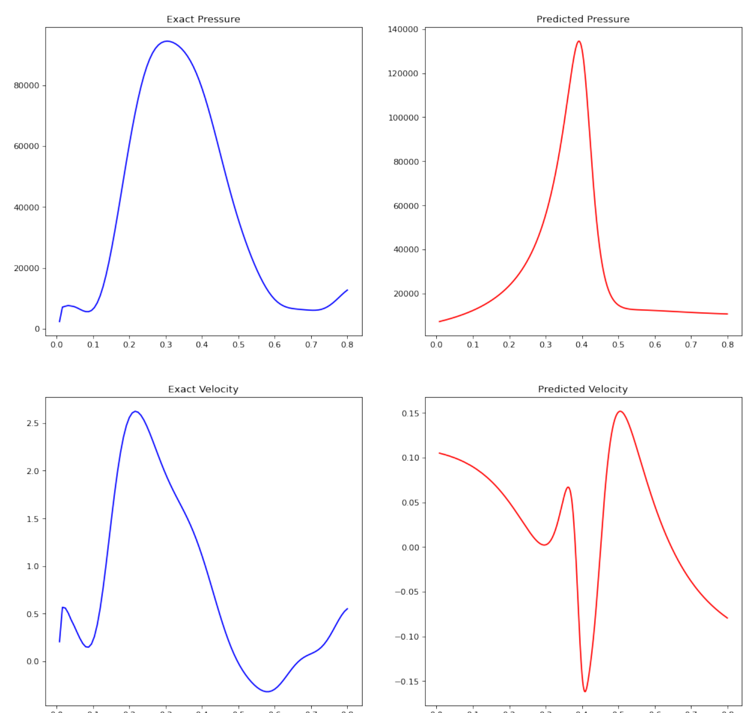

Figure 3: Results: Exact values vs predicted for pressure and velocity

When the obtained model is trained for second time for 40000 epochs are learning rate 1e-04, high over fitting was observed(Figure 4)

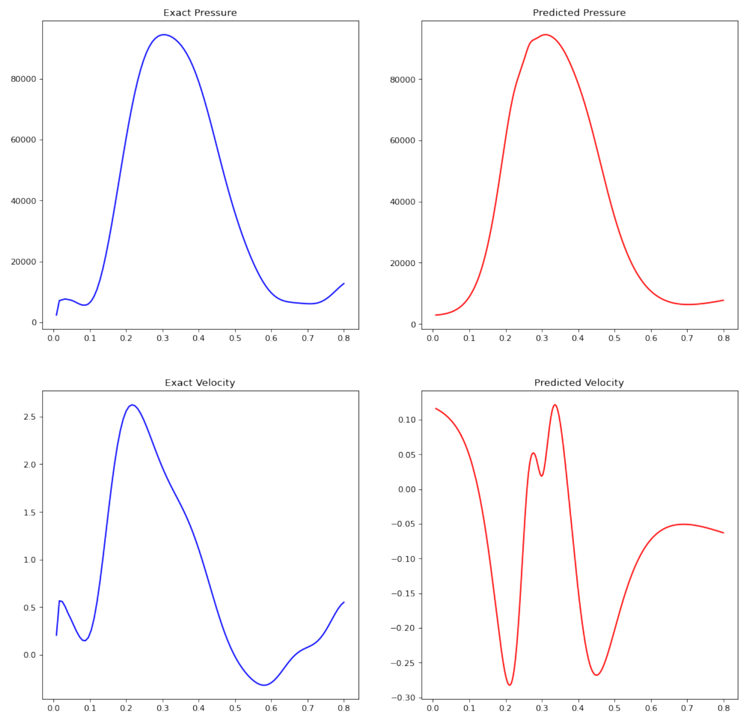

Figure 4: Results: Exact values vs predicted for pressure and velocity

The model almost exactly fitted the pressure data, but very poorly predicted the velocity data. In both the iterations the predictions for model parameters $\lambda 1$ and $\lambda 2$ were not exact but slightly near values.

4 Summary

Referring [1], [2] model coding was done. Most of the code is extracted from the mentioned papers. A base flow is successfully implemented to find the measurement of blood viscosity, velocity and density using only one point pressure data. Though the model is implemented the accuracy is not up to the par. There is still a huge scope of improving model accuracy. The physics of the system defined can be modified to capture more realistic nature of the system. The code is hosted in github at link https://github.com/AHemanthika/BloodViscoVelocityPINN

5 References

[1] 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.

[2] Kissas, G., Yang, Y., Hwuang, E., Witschey, W. R., Detre, J. A., Perdikaris, P. (2020). Machine learning in cardiovascular flows modeling: Predicting arterial blood pressure from non-invasive 4D flow MRI data using physics-informed neural networks. *Computer Methods in Applied Mechanics and Engineering*, 358, 112623.