Exploration on Physics-Informed Neural Networks on Partial Differential Equations (Student Abstract)

Hoa Ta[1], Shi Wen Wong[2], Nathan McClanahan[2], Jung-Han Kimn[2], and Kaiqun Fu[2]

[1] University of California of Irvine
[2] South Dakota State University
hcta@uci.edu, {shiwen.wong,nathan.mcclanahan,jung-han.kimn,kaiqun.fu}@sdstate.edu

Abstract

Data-driven related solutions are dominating various scientific fields with the assistance of machine learning and data analytics. Finding effective solutions has long been discussed in the area of machine learning. The recent decade has witnessed the promising performance of the *Physics-Informed Neural Networks* (*PINN*) in bridging the gap between real-world scientific problems and machine learning models. In this paper, we explore the behavior of *PINN* in a particular range of different diffusion coefficients under specific boundary conditions. In addition, different initial conditions of partial differential equations are solved by applying the proposed *PINN*. Our paper illustrates how the effectiveness of the *PINN* can change under various scenarios. As a result, we demonstrate a better insight into the behaviors of the *PINN* and how to make the proposed method more robust while encountering different scientific and engineering problems.

Introduction

Partial Differential Equations (*PDE*s) are used to model many phenomena and problems related to biology, physics, and engineering. Many *PDE*s are hard to find the analytical solutions. Moreover, with the absence of an extensive amount of training data, it is time-consuming to find numerical solutions for multi-dimensional problems. *Physics-Informed Neural Networks* (*PINN*s) were introduced as deep neural networks that can help solve partial differential equations in numerical methods in an efficient way by taking into account the given laws of physics. The idea of using physics-informed machine learning method to approximate an arbitrary function and its derivatives in Universal Approximate Theorem (*UAT*) has become possible thanks to the advancement of technology. Applying this machine learning network not only helps us approximate the solution of complicated partial differential equations with a small error rate, but also solves them simultaneously in different parameters. In this work, we are learning the behaviors of the Physics-Informed neural network in Burgers' equation, and using different parameters to test out different scenarios that can improve *PINN* efficiency. We also make multiple experiments to test the sensitivity and robustness of the network under different initial conditions and diffusion coefficients.

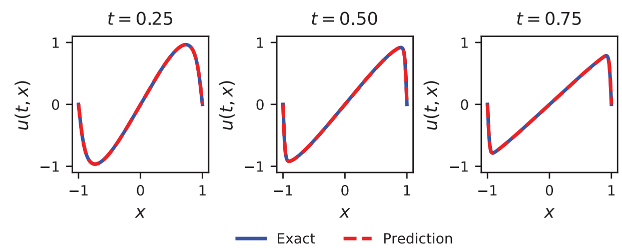

Figure 1: Given initial conditions and boundary conditions, the predictions of the model at $t = \{0.25, 0.50, 0.75\}$.

The scope of this work will contribute to demonstrating the applications of the *PINN*s to biofilm model prediction, fluid mechanics, and some other problems related to partial differential equations.

Problem Setup and Methodology

Given a Burgers' equation $f(t,x) = u_t + \lambda_1 u u_x - \lambda_2 u_{xx}$, with Homogeneous Dirichlet boundary conditions (Schiaffino and Tesei 1982) such that $u(0,x) = G(x) = 0$, $x \in [x_1, x_2]$, $t \in [t_1, t_2]$, $u(t,x_1) = u(t,x_2) = 0$; $\lambda = (1, \lambda_2)$. $G(x)$ denotes the initial conditions for the given burgers equation. We then can generate a matrix of solutions $F(t,x)$ over the range x, t using analytical methods (Basdevant et al. 1986; Cole 1951). Our question is how can we predict the solution of the equation using a neural network when the only pieces of information we have are x, t, and the matrix solution $F(t,x)$. Assume that $Y(t,x,W,b)$ is a neural network that can generate a good approximated solution for $f(t,x)$. Now, our problem can be represented to find the optimal W and b such that $1/N \sum_i |u(t_u^i, x_u^i) - Y(t_u^i, x_u^i; W, b)|^2$ will achieve its minimum value.

Considering that we want to predict the solution for a burgers' equation in this following form: $f(t,x)$=$u_t + uu_x - \lambda_2 u_{xx}$, over the range x, t, with Dirichlet boundary conditions such that $u(0,x) = G(x)$. $G(x) \in \{-sin(\pi x), sin(\pi x), -sin(2\pi x), sin(2\pi x)\}$. We reference the *PINN*s implementation of Raissi but focus on the data-driven solution for a continued time of the burgers equation (Raissi, Perdikaris, and Karniadakis 2019).

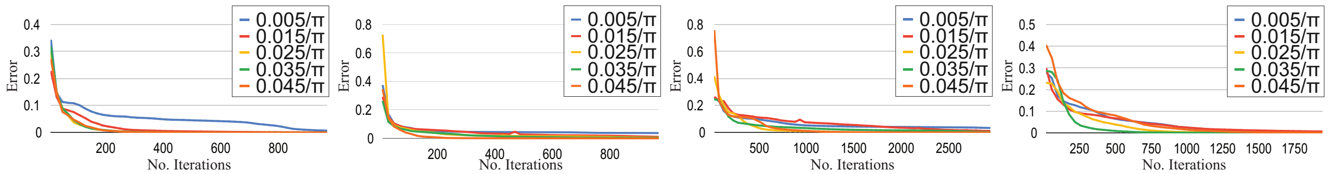

Figure 2: Burgers' equation with four different initial condition $u(0, x) = \{-sin(\pi x), sin(\pi x), -sin(2\pi x), sin(2\pi x)\}$, with the vertical lines showing errors and the horizontal lines showing the number of evaluations

Init. Con.	$-sin(\pi x)$	$sin(\pi x)$	$-sin(2\pi x)$	$sin(2\pi x)$
Error	4.5e-04	3.9e-04	1.0e-02	2.9e-03
Eval.	3773	4790	9022	8106

Table 1: The table compares the performance of the model in four different initial conditions using burgers' equation $u_t + uu_x - (0.045/\pi)u_{xx} = 0, x \in [-1, 1], t \in [0, 1]$

Experiment and Results

In this section, we discuss the experiment setup, sampling methods for the training data, and the experimental results.

Datasets: In order to evaluate the performance of *PINN*, a labeling dataset is pre-calculated with analytical solutions. In order to study the correlations between the hyperparameters (selections of initial conditions and diffusion coefficients) and model performance, we pre-calculated the solutions of four initial conditions and selected ten different diffusion coefficients. Each solution $u \in \mathbb{R}^{256\times100}$, where 256 is the spatial dimension and 100 is the temporal dimension.

Sampling: In terms of picking sample points for the neural network, 100 random points in the boundary set are picked for prediction. The sampling points, which are used to reinforce the law of physics of the partial differential equation including 2 parts: 10,000 collocation points were generated using the Latin Hypercube Sampling method, and the whole boundary data points of the given partial differential equation.

Initial conditions: We exploit the robustness of the model in four different initial conditions: $u(0, x) = \{-sin\pi x, sin\pi x, -sin2\pi x, sin2\pi x\}$, all four initial conditions satisfy the Dirichlet boundary condition. When the initial condition of a *PDE* is changed, the constraint of the Dirichlet boundary condition may not hold.

It is important to note that changing the form of initial conditions causes a great change in the performance of the network, particularly in the number of evaluations of the network and its error rate. The results in Table 1 indicate that the more complicated the initial condition is, the less robust the network becomes.

Diffusion coefficient: We explore ten different diffusion coefficients from $0.005/\pi$ to $0.05/\pi$ for each new initial condition. Within the range of our experiment, we noticed that in general there is an improvement in performance when the diffusion coefficient (λ_2) is increased. We observe that with all ten different diffusion coefficients, the network will be converged eventually at a small error rate. However, with a higher diffusion coefficient, for example $0.045/\pi$, the model will be trained better than $0.005/\pi$ because the error is dropped faster and the network reaches a lower converged point. The results are shown in Figure 2.

Nevertheless, it is noteworthy to mention that when the coefficient is dramatically increased to a big number, we run into an overfitting problem, which makes the performance decline dramatically.

Conclusion

We can use *PINN* to approximate the solution for Burgers' equation. Even though the error rate may be different for different parameters of the equation, the error rate of the network will eventually converge at an acceptable low rate. In our experiment, increasing the diffusion coefficients can improve the robustness of the network; however, as partial differential equations are highly sensitive to small changes, a large increase in diffusion coefficient can cause overfitting, which makes the network's performance decrease. To continue this research we can study the network with other complex initial conditions, use different Partial Differential Equations, study the network with high dimensional data sets or apply some other machine learning techniques to improve the robustness of the *PINN* networks.

Acknowledgements

This research was sponsored by the National Science Foundation (NSF) under Grant No.1950448. We would like to acknowledge with thanks the tremendous support from the NSF and South Dakota State University during our research.

References

Basdevant, C.; Deville, M.; Haldenwang, P.; Lacroix, J.; Ouazzani, J.; Peyret, R.; Orlandi, P.; and Patera, A. 1986. Spectral and finite difference solutions of the Burgers equation. *Computers & fluids*, 14(1): 23–41.

Cole, J. D. 1951. On a quasi-linear parabolic equation occurring in aerodynamics. *Quarterly of applied mathematics*, 9(3): 225–236.

Raissi, M.; Perdikaris, P.; and 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.

Schiaffino, A.; and Tesei, A. 1982. Competition systems with Dirichlet boundary conditions. *Journal of Mathematical Biology*, 15(1): 93–105.