Physics informed machine learning for modeling complex dynamical systems

Introduction

The use of machine learning (ML) techniques to model and predict complicated dynamical systems has dramatically increased over the last ten years. Even while these models have performed remarkably well in settings with a wealth of data, they are essentially indifferent to the physical rules that underlie the behaviour of the system. Because of this, they frequently exhibit inadequate data efficiency, fragility in extrapolation regimes, and a propensity to generate outputs that defy established physical bounds.

Conventional machine learning models are fundamentally statistical approximators that optimize empirical loss functions to map inputs to outputs. Physical systems, on the other hand, are not just statistical entities; they are also subject to conservation laws, symmetries, invariants, and partial differential equations (PDEs). In scientific and technical applications where interpretability and physical consistency are essential, disregarding these principles during training might lead to models that are correct on training data but unreliable when deployed.

A new paradigm known as Physics-Informed Machine Learning (PIML) has emerged as a result of the increasing conflict between the structured, equation-driven character of physical systems and the black-box nature of machine learning. By directly integrating known physics, usually in the form of PDEs, ODEs, or variational principles, into the learning target, PIML aims to integrate data-driven learning with the mathematical framework of physical laws. Models produced using this hybrid approach include:

- Require significantly less data, as they are guided by physical priors,
- Can generalize better to unseen regimes due to structural inductive bias,
- And are guaranteed to obey physical laws, thus increasing trust and interpretability.

In this work, we investigate this paradigm by comparing four different models that are applied to the 1D viscous Burgers' problem, a basic PDE from fluid dynamics. These models are:

- Sparse Identification of Nonlinear Dynamics (SINDy) — a model discovery method that seeks interpretable governing equations from data.
- Physics-Informed Neural Networks (PINNs) — a class of neural networks trained not only on data, but also on the residuals of governing PDEs.
- Neural Ordinary Differential Equations (Neural ODEs) — a framework for learning continuous-time dynamics via differentiable solvers.
- Fourier Neural Operators (FNOs) — deep learning models that learn solution operators to PDEs in function space using spectral representations.

Every model has its own advantages and presumptions. We examine the correctness, physical faithfulness, data efficiency, and interpretability of several approaches using the Burgers' equation as a guide. *In conclusion, we want to clarify if machine learning models can learn physics using the language of physics itself, rather than merely data.*

Burger's equation

We use the one-dimensional viscous Burgers' equation, a classic nonlinear partial differential equation that has long been used as a standard in the study of fluid mechanics, shock formation, and nonlinear wave propagation, as the foundation for our comparative analysis of physics-informed machine learning models.

The following is the 1D viscous Burgers' equation:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \nu\frac{\partial^2 u}{\partial x^2}$$

Where:

- $u(x,t)$ is the velocity field,
- v is the viscosity coefficient

The Burgers equation, in spite of its simplicity, incorporates key aspects of fluid dynamics, including dissipation, shock production, and wave propagation. It is the perfect standard for assessing data-driven and physics-informed models because of its harmony between nonlinearity and diffusion.

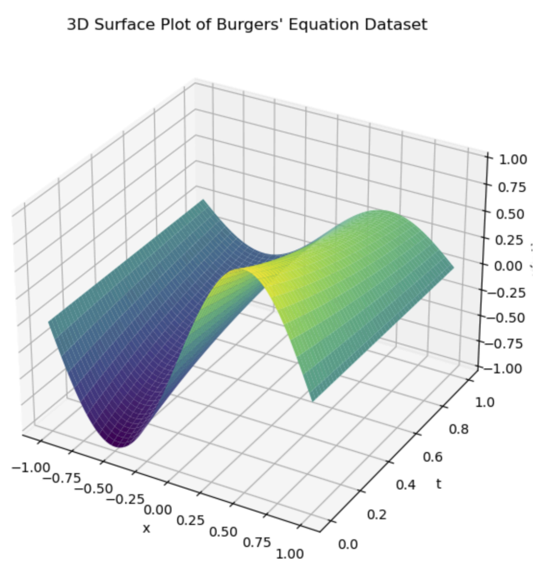

figure1:The velocity field u(x,t) produced by solving the 1D viscous Burgers' equation is shown in a 3D surface map. The surface displays the evolution of the original sine wave over time as a result of viscous diffusion and nonlinear advection.

SINDy(Sparse Identification Of Non-linear Dynamics)

Finding the fundamental governing equations of a dynamical system directly from measurement data is the goal of the data-driven model discovery technique known as SINDy (Sparse Identification of Nonlinear Dynamics).it is a sort of GLM that solves the equation:

$$\dot{\mathbf{x}}(t) = \mathbf{\Theta}(\mathbf{x}(t))\, \boldsymbol{\xi}$$

By assuming that only a small number of terms are active among a large number of potential nonlinear functions, SINDy, in contrast to black-box machine learning models, produces an interpretable mathematical model, usually in the form of an ordinary differential equation (ODE).

The governing dynamics that best explain the observed dataset are represented by the equation that SINDy found. SINDy searches a vast library of possible terms using sparse regression without any prior knowledge of the Burgers' equation. It then determines the smallest combination that can replicate the data's temporal history. Interestingly, the found equation nearly resembles the actual Burgers' equation:

$$\partial u/\partial t = -0.998 \cdot u \cdot \partial u/\partial x + 0.0098 \cdot \partial^2 u/\partial x^2$$

PINN(Physics Informed Neural Network)

Through the direct integration of the underlying physical principles into the neural network's training process, Physics-Informed Neural Networks (PINNs) provide a potent framework for solving partial differential equations. PINNs use automatic differentiation to maintain consistency with the underlying PDE rather than depending on densely labelled data across the domain.

In our setup, we use a fully connected neural network $u(x, t: \theta)$, where θ are the trainable weights, to approximate the solution to the 1D viscous Burgers' equation. The network is trained to minimise a two-part total loss function:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{data}} + \lambda_{\text{PDE}} \cdot \mathcal{L}_{\text{PDE}}$$

In this case, physics loss enforces the Burgers' equation residual, while data loss enforces the initial and boundary conditions:

$$\mathcal{L}_{\text{PDE}} = \frac{1}{N_f} \sum_{i=1}^{N_f} \left| \frac{\partial u}{\partial t}(x_i, t_i) + u(x_i, t_i) \frac{\partial u}{\partial x}(x_i, t_i) - \nu \frac{\partial^2 u}{\partial x^2}(x_i, t_i) \right|^2$$

Physics violations at randomly chosen collocation points (xi, ti) within the domain are penalised by this loss. The network can satisfy the PDE across the domain without explicit labelling since it uses automatic differentiation to generate all necessary spatial and temporal derivatives of $u(x, t{:}\, \theta)$ during training.

The strength of directly integrating physical priors into the learning process is demonstrated by this method, which allows the network to learn a physically consistent approximation of the solution even when there is a lack of data.

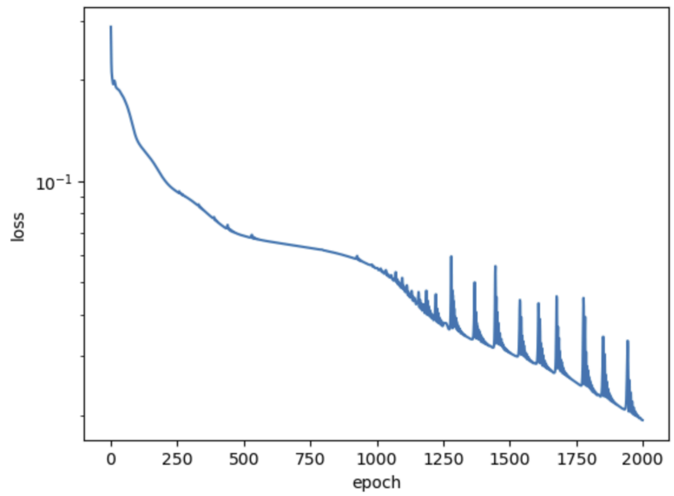

Figure 2: Training Loss Curve over 2000 Epochs

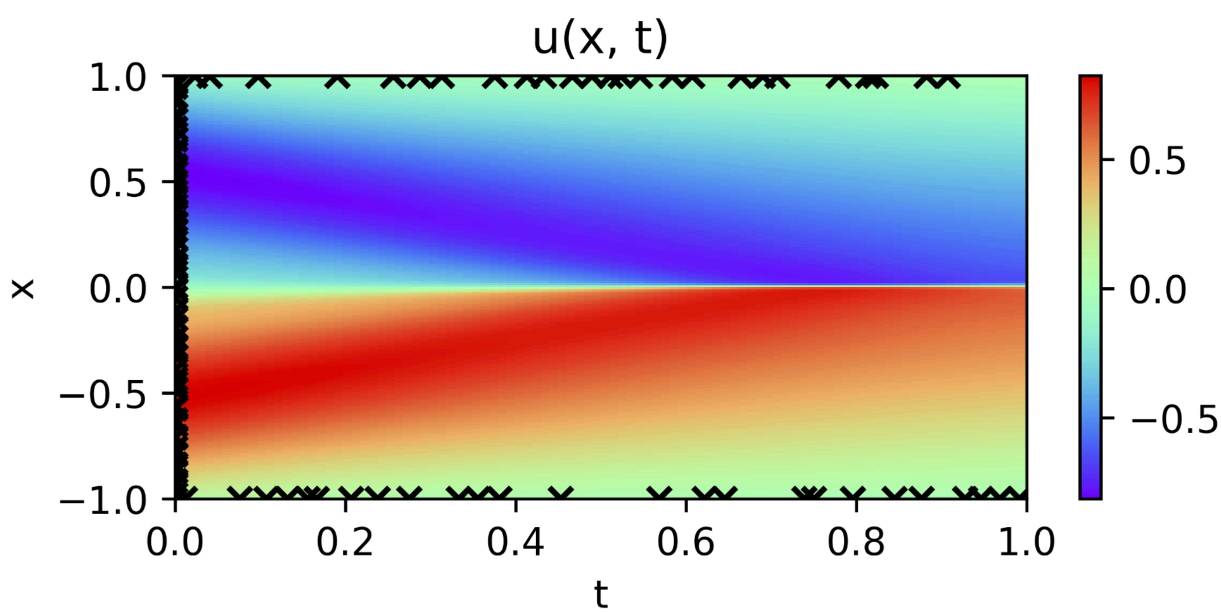

Figure 2: Heatmap of u(x,t) Solution

With x(-1.0 to 1.0) on the y-axis and t (0.0 to 1.0) on the x-axis, the heatmap above depicts PINN's solution to Burgers' equation. It uses a colour gradient from purple (negative) to red (positive) to display $u(x, t)$ values (-0.5 to 0.5). The model's compliance with restrictions is highlighted by the black 'x' markers that denote sampling boundary and initial condition locations. The progression of the shock waves is well depicted in this plot.

Neural Ordinary Differential Equation

A class of models known as Neural Ordinary Differential Equations (Neural ODEs) uses a neural network to parameterise the differential equation itself in order to learn to characterise the time evolution of a system.

Neural ODEs learn the function $f(u, t\colon \theta)$ that defines the dynamics rather than immediately forecasting future values:

$$\frac{du}{dt} = f(u, t; \theta)$$

This loss penalizes deviation between the ODE-simulated trajectory and the observed data at discrete time points ti.

$$\mathcal{L}_{\text{ODE}} = \sum_{i=1}^{N_t} |u_{\text{true}}(t_i) - u_{\text{pred}}(t_i)|^2$$

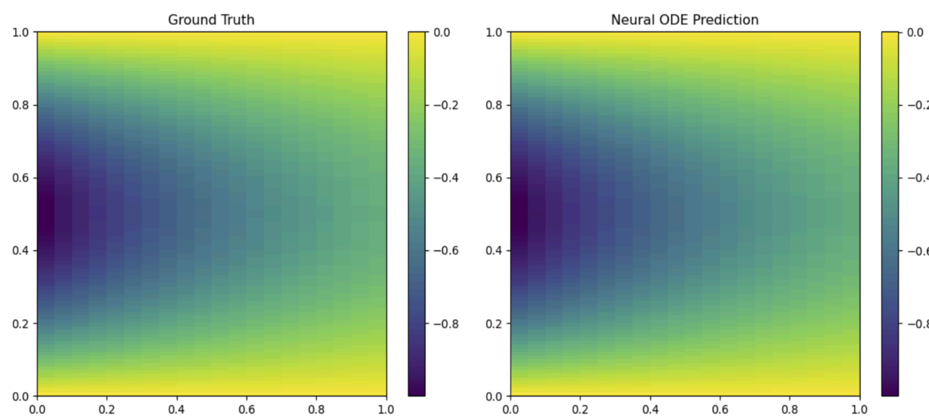

figure:Comparison of neural ODE-predicted solutions of $u(x, t)$ with ground truth. The Neural ODE provides a high-fidelity representation of the system's spatial temporal behaviour.

Fourier Network Operator

A deep learning framework called the Fourier Neural Operator (FNO) was created to discover mappings across infinite-dimensional function spaces. FNOs are designed to learn partial differential equation (PDE) solution operators, or functions that translate input fields (like

initial/boundary conditions) to output fields (like full solution throughout space and time), in contrast to conventional neural networks that deal with finite-dimensional vectors.
The fundamental breakthrough is in the frequency-domain data processing of FNOs. FNOs may effectively capture complicated nonlinear dynamics and long-range spatial dependencies, particularly in systems defined by PDEs, by learning on the spectrum coefficients through Fourier transforms applied to input functions.

In the experiment, FNO model was trained to learn the mapping of:

$$u(x,0) \rightarrow u(x,t)$$

The FNO is trained using a standard supervised learning loss:

$$\mathcal{L}_{\text{FNO}} = \frac{1}{N} \sum_{i=1}^{N} \left| u_{\text{true}}^{(i)} - u_{\text{pred}}^{(i)} \right|^2$$

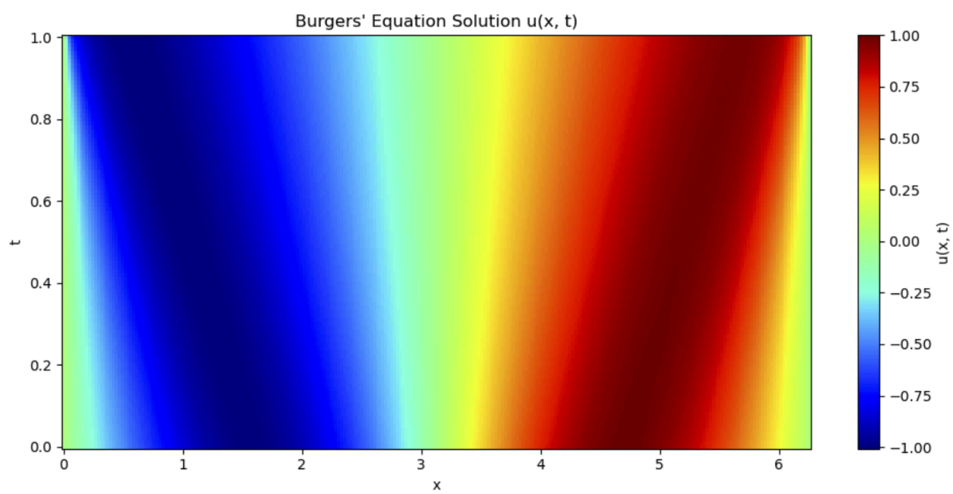

figure:The 1D Burgers' equation u(x,t) is represented graphically as a ground truth solution across time t and space x.

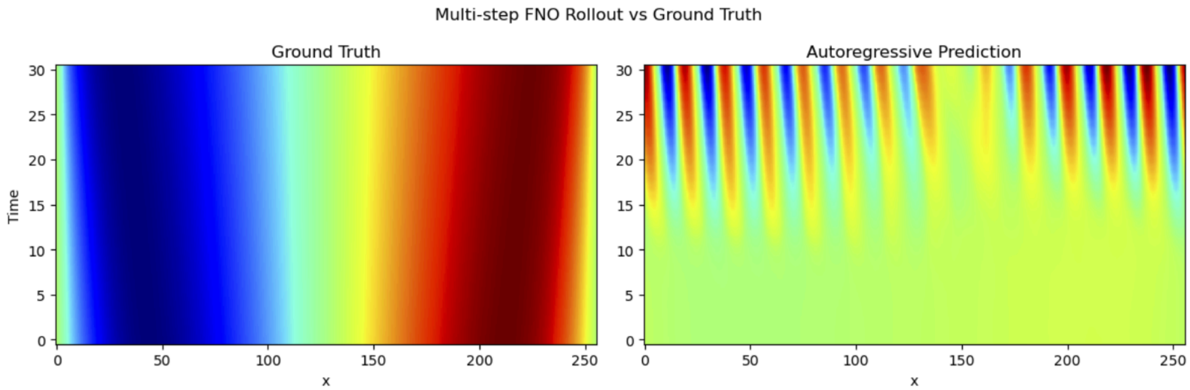

figure:Comparison of the ground truth (left) and FNO autoregressive predictions (right) for the Burgers' equation solution over time in a multi-step rollout.

Conclusion

Four physics-informed methods—PINNs, SINDy, Neural ODEs, and FNOs—applied to the 1D Burgers' equation are presented in this paper. Different approaches are taken by each model to capture dynamics: PINNs directly incorporate the PDE structure into the loss function, SINDy recovers sparse symbolic representations, Neural ODEs use learnt dynamics to model continuous-time evolution, and FNOs use spectral convolution to learn operators. Despite having different goals and designs, all techniques successfully restore important aspects of the underlying system. This investigation highlights the flexibility of physics-based machine learning in simulating nonlinear PDEs and offers information on how physical restrictions can be included into data-driven models to enhance generalisation and interpretability, particularly in situations when analytical solutions are unfeasible.

References

- **Fourier Neural Operator for Parametric Partial Differential Equations**

  Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, Anima Anandkumar

- **Neural Ordinary Differential Equations**

  Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, David Duvenaud

- **Physics Informed Deep Learning (Part I): Data-driven Solutions of Nonlinear Partial Differential Equationshttps://arxiv.org/abs/1711.10561**

  Maziar Raissi, Paris Perdikaris, George Em Karniadakis

- **Discovering governing equations from data: Sparse identification of nonlinear dynamical systems**

Steven L. Brunton, Joshua L. Proctor, J. Nathan Kutz

- A Review of Data-Driven Discovery for Dynamic Systems
  Joshua S. North1,*, Christopher K. Wikle2, and Erin M. Schliep3

Name - Dhruv Kulshrestha,
PRN - 22070127018