pyCLINE: Python implementation of CLINE method for discover of hidden nullcline structures in oscillatory dynamical systems

14 February 2025

Summary

Many important processes in physics, biology, chemistry and engineering are dynamical, just to mention some examples such as the orbiting of planets around the sun, the preiodic oscillations of the cell cycle, the Belousov-Zhabotinsky reaction or movements of fluids. Therefore, understanding and describing dynamical systems is immenently important. The study of dynamical systems has been done by deriving mathematical models in the form of differential equations. This requires a mix of analysis of gathered data, established modelling principles and scientific intuition. However, recent technological developments in the field of machine learning have changed how dynamical systems can be studied: Black-box methods allow to recreate a dynamical system and using perturbation study it overall dynamical behavior. Such methods include recurrent neural networks, To tackle this challenge, many data-driven or also called machine learning approaches have been suggested in recent years, that allow to derive symbolic or form-free models directly from measured data, reservoir computing or neural differential equations. White-box methods aim to derive symbolic differential equations directly from measured data by selecting a subset of terms from set of suggested terms (mechanisms). Such methods include, Sparse Identification of nonlinear dynamics (SINDy)(Brunton, Proctor, and Kutz 2016), Nonlinear Autoregressive Moving Average Model with Exogenous Inputs (NARMAX)(Billings 2013) or Symbolic Regression (SR)(Cranmer et al. 2019).

The disadvantage of black-methods is their lack of interpratability of underlying mechanisms and of white-box methods their high data quality requirements (Prokop and Gelens 2024). Therefore, so-called grey-box methods aim to combine the strength of black-methods to handle large, structured data sets and still provide interpertable results, such as Physics-Informed neural networks (PINN)(Karniadakis et al. 2021), biology-informed neural networks (BINNs) (Lagergren et al. 2020) or Universal Differential Equations (Rackauckas et al. 2020) and many more.

However, most of the existing methods focus on forecasting in order to determine a model

that is able to adequately describe the temporal evolution of a system. In contrast to that, we have developed a new grey-box method called CLINE (**C**omputational **L**earning and **I**nference of **N**ullcline **E**quations) (Prokop et al. 2025), that instead focuses on identifying the underlying static information in the phase space such as the nullcline structure. Knowledge of the nullcline structure has many advantages (Prokop, Frolov, and Gelens 2024):

- Nullclines fully describe the dynamical systems behavior and therefore provide more information then just the time series
- When the structure of nullclines is known, the symbolic equation can be derived using symbolic model identification methods, but applied to a problem of signficantly lower complexity then time series data
- Identifying the structure without a predefine set of candidate terms (e.g. in form of a library) but model-free allows to avoid implementation of false bias in the model formulation

Methodology

The main aspects of the CLINE method are explained in (Prokop et al. 2025), nevertheless we provide a brief explanation of the method. In order to identify nullclines for a set of ordinary differential equations (ODEs), we have to set the derivative to 0:

$u_t = f(u,v) →u_t = f(u,v)=0, \ v_t = g(u,v) →v_t = g(u,v)=0. $

The functions of f and g are not know *a prior*. However, to learn the functions we can reformulate the nullcline equations to,

$u = f^{-1}(v, u_t)$ or $v = f^{-1}(u, u_t)$
$u = g^{-1}(v, v_t)$ or $v = g^{-1}(u, v_t)$.

Now we have to learn the inverse equations f^{-1} and g^{-1}, which can be expressed as a feed-forward neural network with inputs u and u_t, to learn v. Here, u and v are time series data and u_t the derivative of the u component, on which the neural network is trained. After training,we provide a set of u together with $u_t = 0$ (requirement for a nullcline) as inputs and learn the corresponding values of v that describe $u_t = f(u, v) = 0$. As a result, we learn the structure of a nullcline in the phase space u, v, to which other white-box methods can be applied to learn the symbolic equations, yet on deciseively simpler optimization problem then simply time series data.

Statement of need

`pyCLINE` is a Python package that allows to easily set up and use the CLINE method as explained and shown in (Prokop et al. 2025). It is based on the Python torch implementation `pyTorch` and allows to quickly implement the identification of nullcline structures from simulated or measured time series data. The implementation of `pyCLINE` allows to generate examplary data sets from scratch, correctly prepare data for training and set up the feed forward neural network for training.

`pyCLINE` was designed to be used by researchers experienced with the use of machine learning or laymen that are interested to apply the method to either different models or measured data. This allows for simple and fast implementation in many fields that are interested in discovering nullcline structures from measured data, that can help develop novel or provide proof for existing models of dynamic (oscillatory) systems.

Usage

The `pyCLINE` package includes three main modules (see Figure 1):

- `pyCLINE.generate_data()`: This module generates data whifh has been used in (Prokop et al. 2025) with additionally many more models that can be found under `pyCLINE.model()`.
- `pyCLINE.recovery_methods.data_preparation()`: Splits and normalizes that data for training, with many more features for the user to change the data.
- `pyCLINE.recovery_methods.nn_training()`: Is the `pyTorch` implementation that sets up the model and trains it.

The `pyCLINE.model()` currently includes a set of different models:

- FitzHugh-Nagumo model
- Bicubic model
- Gene expression model
- Glycolytic oscillation model
- Goodwin model
- Oregonator model
- Lorenz system
- Roessler system
- Delay oscillator (self-inhibitory gene)

Some of the models are three-dimensional and can be used to further study the limitations of the method, when applied to higher dimensional systems

For a better understanding the method and a simpler implementation, we also provide `pyCLINE.example()` which contains full examples of how `pyCLINE` can be used. Here, `pyCLINE.example()` can be used to generate prediction data for four systems: The FitzHugh-Nagumo model with time scale separation variable $\varepsilon = 0.3$ (`FHN`), the Bicubic model (`Bicubic`), gene expression model (`GeneExpression`) and the delay oscillator model (`DelayOscillator`).

References

Billings, Stephen A. 2013. *Nonlinear System Identification.* Chichester, UK: John Wiley & Sons, Ltd. https://doi.org/10.1002/9781118535561.

Brunton, Steven L, Joshua L Proctor, and J Nathan Kutz. 2016. "Discovering Governing Equations from Data by Sparse Identification of Nonlinear Dynamical Systems." *Proceedings of the National Academy of Sciences* 113 (15): 3932–37. https://doi.org/10.1073/pn

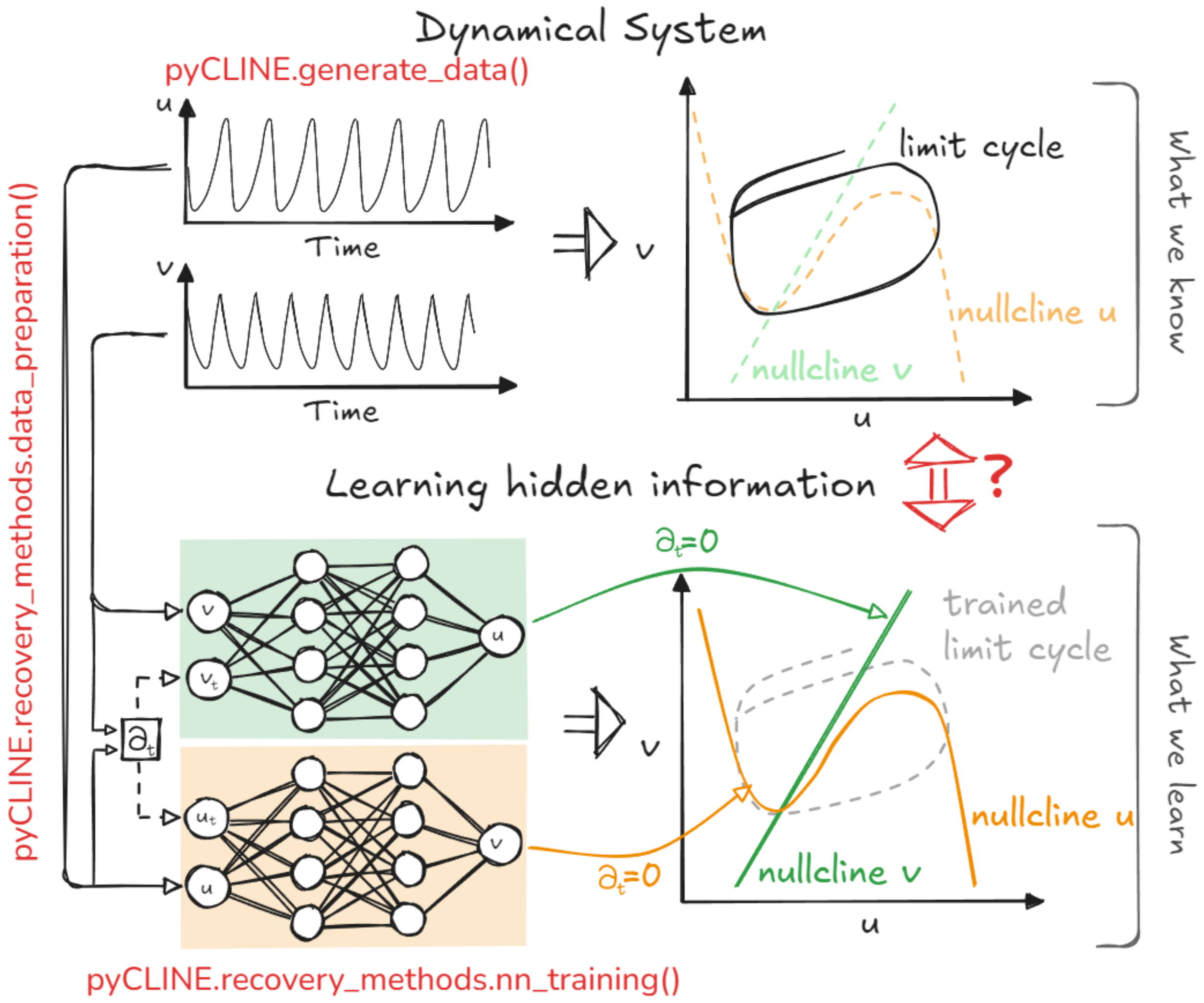

Figure 1: The method CLINE explained by using Figure 1 from (Prokop et al. 2025). In red the main modules of the `pyCLINE` package are shown

as.1517384113.

Cranmer, Miles D., Rui Xu, Peter Battaglia, and Shirley Ho. 2019. "Learning Symbolic Physics with Graph Networks." *ArXiv*, September. https://arxiv.org/abs/1909.05862.

Karniadakis, George Em, Ioannis G. Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. 2021. "Physics-Informed Machine Learning." *Nature Reviews Physics* 3 (6): 422–40. https://doi.org/10.1038/s42254-021-00314-5.

Lagergren, John H., John T. Nardini, Ruth E. Baker, Matthew J. Simpson, and Kevin B. Flores. 2020. "Biologically-Informed Neural Networks Guide Mechanistic Modeling from Sparse Experimental Data." Edited by Inna Lavrik. *PLOS Computational Biology* 16 (12): e1008462. https://doi.org/10.1371/journal.pcbi.1008462.

Prokop, Bartosz, Jimmy Billen, Nikita Frolov, and Lendert Gelens. 2025. "Uncovering Hidden Nullcline Structures Behind Time Series Data Using Neural Networks." https://arxiv.org/abs/XXXX.XXXXX.

Prokop, Bartosz, Nikita Frolov, and Lendert Gelens. 2024. "Enhancing Model Identification with SINDy via Nullcline Reconstruction." *Chaos: An Interdisciplinary Journal of Nonlinear Science* 34 (6): 063135. https://doi.org/10.1063/5.0199311.

Prokop, Bartosz, and Lendert Gelens. 2024. "From Biological Data to Oscillator Models Using SINDy." *iScience* 27 (4): 109316. https://doi.org/10.1016/j.isci.2024.109316.

Rackauckas, Christopher, Yingbo Ma, Julius Martensen, Collin Warner, Kirill Zubov, Rohit Supekar, Dominic Skinner, Ali Ramadhan, and Alan Edelman. 2020. "Universal Differential Equations for Scientific Machine Learning," January.