

Learning state variables for physical systems

The problem of automatically determining state variables for physical systems is challenging, but essential in the modeling process of almost all scientific and engineering processes. A deep neural network-based approach is proposed to find state variables for systems whose data are given as video frames.

Boris Kramer

The discovery of natural and physical laws from experimental data via the scientific method has challenged scientists for almost five centuries. This process has been rejuvenated by recent advances in machine learning alongside a tremendous increase in computing power, which together enable new avenues for scientific model discovery: advanced algorithms can find physical laws by sifting through large amounts of data^{1–4}. Once found, the governing equations describe functional relationships between physically meaningful variables. These algorithms are commonly fed labeled and measured data of physical variables in rather low dimensions. The preceding problem, namely to determine which variables are necessary to model a physical system, has seen much less progress, but is equally important for scientific discovery. In this issue of *Nature Computational Science*, Chen et al.⁵ present an approach to discover a set of variables that describes a physical process that is given in the form of a sequence of video frames.

Computational discovery of physical laws from data has a history going back to the 1970s when computers started to have the necessary processing power to implement somewhat intensive search and regression algorithms. The early work of Langley and

co-authors produced the BACON software¹ that was able to determine physical laws — albeit from low-dimensional states — and to identify some functional relationships between independent and dependent variables. The identification process tested selective hypotheses, that is, it was looking for common heuristic laws such as constant and linear relationships, multiplicative and inverse correlations, and so forth. BACON was also able to add additional variables to an existing set, if it so discovered that a certain functional relationship existed. In 2009, Schmidt and Lipson² proposed a genetic algorithm that would learn physical laws, including conservation laws of Hamiltonian and Lagrangian nature, from experimental data. Their work would correctly identify many known physical laws and became inspirational to future algorithmic developments. However, the employed genetic algorithms could be unstable and quite sensitive to the data (and noise therein). Moreover, they did not enforce what Occam's razor hypothesized: that natural laws often arise in a simple form. Brunton, Proctor and Kutz⁴ used that very parsimonious principle to equip their symbolic regression algorithm with sparsity promoting regularization, which has been demonstrated to learn a variety

of engineering and physical principles. Recently, a multidimensional symbolic regression algorithm with additional physics-inspired heuristics⁴ correctly identified all 100 equations in the *Feynman Lectures on Physics*. Yet, despite all this progress, none of these approaches tackled the 'upstream' problem of identifying the proper physical variables to describe these systems. While alternative sets of physical variables to determine a dynamical system can be found manually⁶, algorithmic approaches are sorely needed.

While other model identification methods typically rely on sensor data, which directly measure a specific variable, the proposed approach by Chen et al. generalizes to any physical system and avoids expensive experimental setups. The key of the algorithm is to leverage the structure of deep neural networks, particularly of encoder–decoder type. The authors trained the neural network to correctly predict given video frames and then they use a strategy to find out how many internal neurons — which the authors call the intrinsic dimension — the network needs for accurate predictions. The intrinsic dimension informs the number of state variables needed, and after an additional network is leveraged, results in a set of variables to describe the

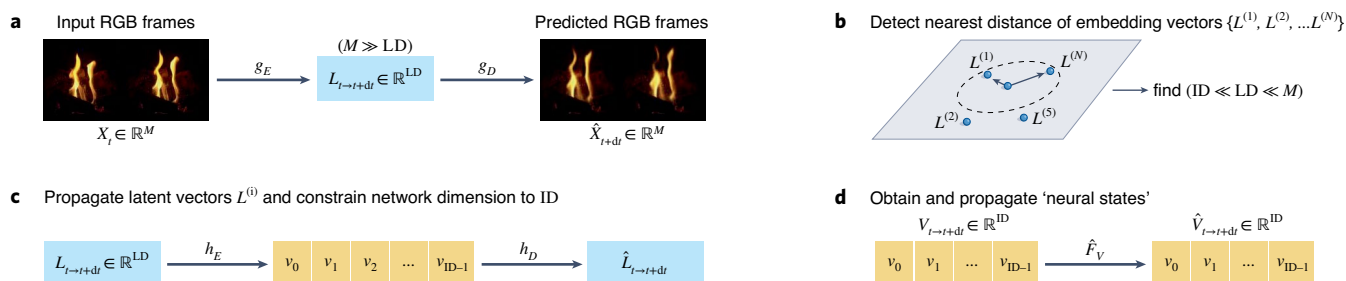


Fig. 1 | Workflow for the designed data-driven framework that finds state variables. **a**, The first step determines an encoder (g_E)–decoder (g_D) network that propagates the RGB images by one timestep. The latent vectors of the forward map are denoted L_{t-t+dt} . **b**, The next step detects the intrinsic dimension (ID), which is much smaller than the latent dimension (LD) and the input size (M), that is, $M \gg LD \gg ID$. The intrinsic dimension is found through a nearest-neighbor-type argument in the nonlinear manifold of latent vectors $L^{(i)}$. **c**, A new encoder (h_E)–decoder (h_D) network is fitted to the latent vectors $L^{(i)}$, where the internal dimension is fixed to ID. The internal variables v_i are thus interpreted as the 'neural states', that is, the minimal number of internal states needed to predict the model. **d**, The neural states are propagated through the learned dynamics $\hat{F}_V$.

evolution of the physical system. The authors demonstrated the applicability of their approach by successfully predicting system dynamics in a variety of dynamical systems including chaotic kinematics (rigid double pendulum, elastic double pendulum, swing stick), nonlinear waves (reaction–diffusion system), multi-phase flow (lava lamp), aeroelasticity (air dancer) and combustion (flame dynamics).

More specifically, in the framework, Chen et al.⁵ first fit a typical encoder–decoder architecture (g_E and g_D , as depicted in Fig. 1a) to the RGB (red, green, blue) image data frames, to obtain a forward map that advances the current frame by one time step. In the next stage, they use the latent vector of this encoder–decoder architecture to estimate the intrinsic dimension (Fig. 1b) — that is, the required number of state variables — of the system via the Levina–Bickel algorithm. Once the number of state variables is estimated, the authors determine the actual variables to describe the system — they term them neural state variables —

noting that those are not unique. This step of finding neural state variables is key to their algorithm as it effectively introduces another encoder–decoder architecture (h_E and h_D , as depicted in Fig. 1c) for the latent vectors and fixes the latent dimension of the second encoder–decoder network (Fig. 1c) to the identified intrinsic dimension. This gives them a set of neural state variables, which they then advance through the system dynamics (Fig. 1d) that were learned at the level of the RGB image data.

The identification of physical variables from data of dynamical systems is very challenging and has lacked automation and computationally tractable procedures. Chen et al.⁵ present a framework that will help advance the field and close an important gap. However, it is important to note that the discovered variables do not have any units, and may not be interpreted as physical variables (such as pressure, temperature, and velocity) per se. At the moment, intuitive meaning must still be assigned by an application expert. It will be interesting

to see their present work stimulate more research in this direction where, if possible, a complete set of physical variables with units could be learned from RGB data. 

Boris Kramer  

Department of Mechanical and Aerospace Engineering, University of California San Diego, La Jolla, CA, USA.

 e-mail: bmkramer@ucsd.edu

Published online: 25 July 2022

<https://doi.org/10.1038/s43588-022-00283-4>

References

1. Bradshaw, G. F., Langley, P. W. & Simon, H. A. *Science* **222**, 971–975 (1983).
2. Schmidt, M. & Lipson, H. *Science* **324**, 81–85 (2009).
3. Brunton, S. L., Proctor, J. L. & Kutz, J. N. *Proc. Natl Acad. Sci.* **113**, 3932–3937 (2016).
4. Udrescu, S.-M. & Tegmark, M. *Sci. Adv.* **6**, eaay2631 (2020).
5. Chen, B. et al. *Nat. Comput. Sci.* <https://doi.org/10.1038/s43588-022-00281-6> (2022).
6. Qian, E., Kramer, B., Peherstorfer, B. & Willcox, K. *Physica D* **406**, 132401 (2020).

Competing interests

The author declares no competing interests.