Guided Structural Organisation of Latent Variables Using Artificial Intelligence

Simon Watson Micheal Harré

A challenging problem in non-linear dynamics is the identification of the structural equations that can accurately replicate system dynamics from data. Some of the most popular techniques commonly used include: nonlinear auto-regressive models [1], dynamic mode decomposition [6], genetic programming [5] amongst many others. The approach considered here, which has shown considerable promise on chaotic dynamical systems, is the artificial intelligence (AI) developed by Champion et al [3] that uses a sparse regression approach in order to extract the structural equations of the latent variables directly from data. In this approach, the form of the data interacts with a structural 'library' to guide the development of the system. Using ideas from information theory, this work develops an extension to the original methodology [2].

The methodology is called SINDy (Sparse Identification of Non-linear Dynamics) [2] and its goal is to discover (or recover) a dynamical system of the form:

$$\frac{d}{dt}\mathbf{x}(t) = f(\mathbf{x}(t)) \tag{1}$$

from a given time-series dynamic: $\mathbf{x}(t) = [x_1(t), x_2(t), \ldots, x_n(t)]^T \in \mathbb{R}^n$. It is assumed that the dynamics of $f(\mathbf{x}(t))$ can be sparsely represented based on a library of candidate functions of the form: $\mathbf{\Theta}(\mathbf{x}) = [\Theta_1(\mathbf{x}), \Theta_2(\mathbf{x}), \ldots, \Theta_p(\mathbf{x})]$, for example: $\Theta_i(\mathbf{x}) \in \{x_1, x_2, \dot{x}_1, \dot{x}_2, x_1^2, x_1x_2, \sin(x_1), \ldots\}$. SINDy performs a sparse regression on this library, by which we mean it identifies a small set of regression weights for the $n \times p$ matrix $\mathbf{\Theta}(x)$ that minimises a loss function. The resultant approximation to equation 1 is of the following form:

$$\frac{d}{dt}x_k(t) = f(x_k(t)) \approx \mathbf{\Theta}(\mathbf{x})\xi_k \tag{2}$$

in which the ξ_k terms are vectors of regression weights over the library of terms $\mathbf{\Theta}(x)$. From this it can be seen that the elements of the library $\mathbf{\Theta}(x)$ act as constraints on the possible structural forms that the approximation to $f(\mathbf{x}(t))$ can take and thereby this library 'guides' the unsupervised organisation of the SINDy AI in replicating the original data. The approach has been shown in the Champion et al paper [3] to be resilient to significant experimental noise.

In order to establish the correctness of the implementation in this work, results were produced that replicated the canonical systems Brunton et al first tested including the Lorenz system, a 2D reaction–diffusion system, and a 2D spatial representation (synthetic video) of the nonlinear pendulum. These will be discussed in the presentation in order to illustrate the interpretation of the results in non-trivial examples. Below are the plots of the 1-dimensional non-linear pendulum $\ddot{x} = -\sin(x)$ that is the base case test of the model in the paper [3].

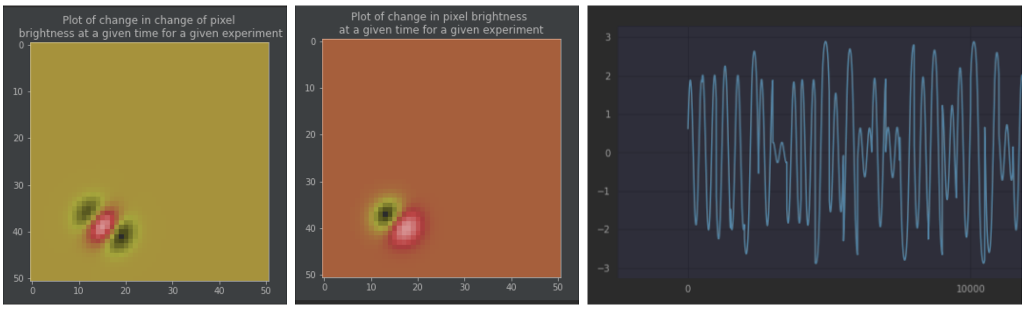

Figure 1: Image of the velocity and acceleration of a point pendulum system along with latent variable z. This variable is used in the manufacture of 'experimental data' and as a driver of the pendulum motion, it is the variable that the SINDy autoencoder must ultimately recover in a successful experiment.

Building on the original work of Champion et al., this work adapts the loss function in the original SINDy Autoencoder system using an information theoretic approach. Rather than the original (modified) L^2 (Euclidean norm) reconstruction loss function [3], an information theory-based approach will be taken, using a Cross Entropy loss $\mathcal{L}_{CE}$ [4].

The autoencoder portion of the model is a neural network with hidden layers that progressively contain less nodes than the input layer in an 'encoding' network, $\varphi(\mathbf{x})$. The second 'decoding' network, $\psi(\mathbf{z})$, is the inverse, taking input $\mathbf{z}$ from the few nodes on the encoding network output and then feeding them forward through layers with progressively more nodes up to the original number of inputs nodes, outputting the reconstructed input $\mathbf{x}'$. The value of the model comes from using the difference between $\mathbf{x}$ and $\mathbf{x}'$ to train the system to learn a stable latent variable representation of the system $\mathbf{z}$. In the case of the SINDy autoencoder, the input contains both the state $\mathbf{x}$ and its derivative $\frac{\mathbf{dx}'}{\mathbf{dt}}$. Using the SINDy methodology introduced in (1) and (2) and noting that $\mathbf{\Xi}$ is the full set of regression weights containing the subset ξ_k introduced in (2):

$$\frac{dx'}{dt} = (\nabla_z \psi(\mathbf{z}))(\mathbf{\Theta}(\mathbf{z}^T)\mathbf{\Xi}) \tag{3}$$

$$\mathcal{L}_{CE} = \sum_X p\left(\frac{dx}{dt}\right)\log\left(p\left(\frac{dx'}{dt}\right)\right) \tag{4}$$

This function will then guide self organisation in the AI as it learns to extract the latent variables that drive non-linear phenomena such as phase transitions.

In exploring information theoretic adaptions of the SINDy auto encoder approach, opportunities to obtain previously intractable processes are created.

- Can we extract/construct latent variables from real world 'noisy' systems (such as financial markets) which play a role equivalent to temperature in the Ising model?
- Do those recovered latent variables allow us to understand dynamic systems in terms of classical phase transitions and can we form models that can anticipate the nature of subsequent events, for example like 'crashes' in economics?
- If more real world dynamical systems can be understood in terms of formal phase transitions, can we use that broader understanding to build a taxonomy of phase transition modes, for example allowing us to differentiate between the 'flash crash' of 2011 and the US government bailout fail in financial crisis of 2007?

Machine learning enables the use of powerful self supervised processes to drive structural organisation of models. These models have the potential to drive both new understanding dynamical systems and drive new insight by re-framing them in terms of well established systems in other domains.

References

[1] Lida Mercedes. Barba Maggi. "Multiscale Forecasting Models". eng. In: (2018).

[2] Steven L. Brunton, Joshua L. Proctor, and J. Nathan Kutz. "Discovering governing equations from data by sparse identification of nonlinear dynamical systems". In: *Proceedings of the National Academy of Sciences* 113.15 (2016), pp. 3932–3937. ISSN: 0027-8424.

[3] Kathleen Champion et al. "Data-driven discovery of coordinates and governing equations". eng. In: *Proceedings of the National Academy of Sciences - PNAS* 116.45 (2019), pp. 22445–22451. ISSN: 0027-8424.

[4] D. R. Cox. "The Regression Analysis of Binary Sequences". eng. In: *Journal of the Royal Statistical Society. Series B, Methodological* 20.2 (1958), pp. 215–242. ISSN: 0035-9246.

[5] "Genetic programming : an introduction on the automatic evolution of computer programs and its applications". eng. In: (1998).

[6] Joshua L. Proctor, Steven L. Brunton, and J. Nathan Kutz. "Dynamic mode decomposition with control". eng. In: *SIAM journal on applied dynamical systems* 15.1 (2016), pp. 142–161. ISSN: 1536-0040.