16th World Congress in Computational Mechanics (WCCM)
21-26 July 2024, Vancouver, Canada
A. Korobenko, M. Laforet, S. Prudhomme, R. Vaziri (Eds)

USING ARTIFICIAL NEURAL NETWORKS FOR SYMBOLIC REGRESSION

NIBODH BODDUPALLI[1], TIM MATCHEN[2] AND JEFF MOEHLIS[3]

[1] Department of Mechanical Engineering
University of California, Santa Barbara
Santa Barbara, CA 93106
nibodh@ucsb.edu

[2] Department of Mechanical Engineering
University of California, Santa Barbara
Santa Barbara, CA 93106
timothy.matchen@proton.me

[3] Department of Mechanical Engineering
University of California, Santa Barbara
moehlis@ucsb.edu, http://sites.me.ucsb.edu/~moehlis/

Key words: System Identification, Symbolic Regression, Nonlinear Systems, Machine Learning

Abstract. Mathematical models allow researchers to understand, analyze, and predict the behavior of systems of physical, biological, and technological interest, and are required for many techniques from dynamical systems and control theory to be used. Unfortunately, it is often impossible to derive mathematical models from first principles, and in such cases system identification is a powerful tool which can be used to deduce models from observed data. Many existing system identification techniques require pre-specification of a dictionary of possible terms in a mathematical model, limiting their ability to give models with the nonlinearities which can arise for biological and other complex systems. We present a methodology which overcomes this limitation by dynamically generating the terms in a model with the necessary complexity and nonlinearity to accurately describe a system's dynamics. This uses a multilayered, operation-based symbolic regression approach, with the capacity to learn combinations and compositions of operations by training artificial neural networks. Our approach provides a powerful alternative to genetic programming strategies for symbolic regression, and can exploit many of the attractive features of artificial neural networks such as a straightforward learning strategy.

1 INTRODUCTION

The importance of mathematical models for systems of physical, biological, and technological interest cannot be overstated [1]. Such models are sets of equations describing some number of variables or observables, and are often given in terms of ordinary or partial differential equations. Mathematical models allow us to understand, analyze, and predict the behavior of systems, and are required for many powerful techniques from dynamical systems

and control theory to be used. Ideally, mathematical models are derived from first principles such as Newton's laws or Maxwell's equations, giving models which respect relevant physical constraints and conservation laws. Much of the historical development of physics, chemistry, and engineering has been based on such "white box" models. Unfortunately, it is often impossible to derive mathematical models from first principles.

With this in mind, there has been much recent interest in developing an automated procedure which is able to deduce complex evolution equations from data. This is a problem in *system identification* - the development of mathematical models from observed data [2]. Models obtained from system identification often rely on *fixed dictionary methods*, which have pre-specified terms of particular forms, with corresponding coefficient values learned from data. Mathematically, fixed dictionary approaches find a model

$$\frac{dx}{dt} = \sum a_i \phi_i(x) \tag{1}$$

where $\{\phi_i(x)\}$ is a pre-specified set of functions, and the a_i's are determined from the data.

Another approach to system identification is symbolic regression, which we view as a *generative dictionary method.* Here one only pre-specifies a set of primitive operations and functions, and a dictionary of possible model terms is generated from combinations and compositions of these primitives. In particular, letting $S_k(x)$ be a *k*-dependent subset of all possible functions of *x* generated by the primitive operations and functions, we seek to find a model

$$\frac{dx}{dt} = F(S_K \circ \cdots \circ S_2 \circ S_1) \tag{2}$$

Here the composition $\circ$ is interpreted in an element-wise manner. Historically, such methods learn models from observed data by randomly combining various terms and operations, and using genetic programming to "mutate" the candidate solutions according to a fitness-weighted selection mechanism [3-5].

As an alternative, we have recently developed a novel neural network framework for symbolic regression [6], in which the generative dictionary is built up from linear combinations, polynomial combinations, simple products, and common operators such as $sin\,(\cdot)$, $sgn(\cdot)$, and $e^{(\cdot)}$. The breadth of the network determines the number of allowable coefficients in the generated functions, and the depth determines the levels of allowed compositions. To obtain models with a manageable number of terms, we use a combination of novel sparsity-inducing regularization terms and the Akaike Information Criterion (AIC) [7] for model selection. The structure of our network is shown in Figure 1, which illustrates the stacks which correspond to subsequent levels of function composition. Each stack contains operational layers as shown in Figure 2, which shows the primitives which are used to generate terms for the model. A powerful feature of our approach is that, unlike fixed dictionary methods, it does not require pre-specification of the exact terms which describe the system's dynamics. We call our method SymANNTEx (pronounced as "semantics"), for Symbolic, Artificial Neural Network-Trained Expressions. More detail on the SymANNTEx network

architecture, operational layers, regularization terms, and hyperparameters is available in [3].

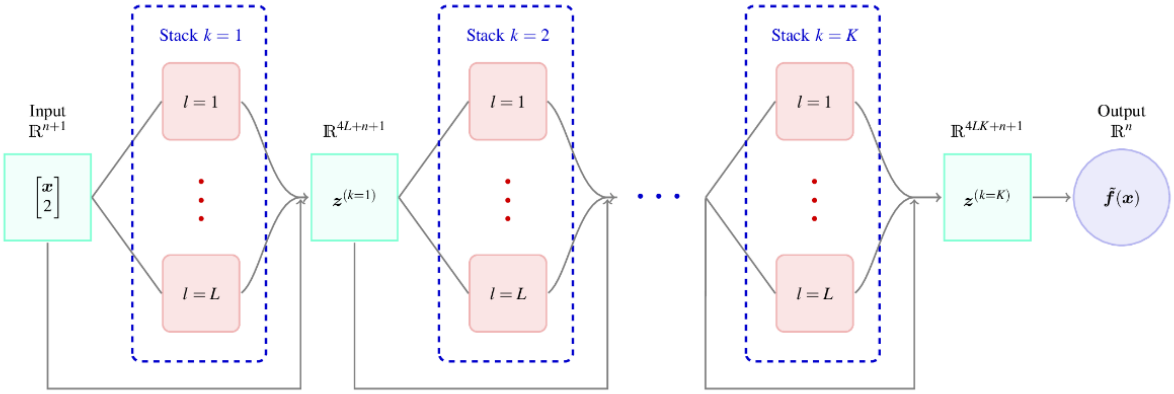

Figure 1: SymANNTEx network architecture: K stacks, with L operational layers within each stack. The stack structure allows compositions of primitive functions to build up more complicated expressions. More detail is available in [6].

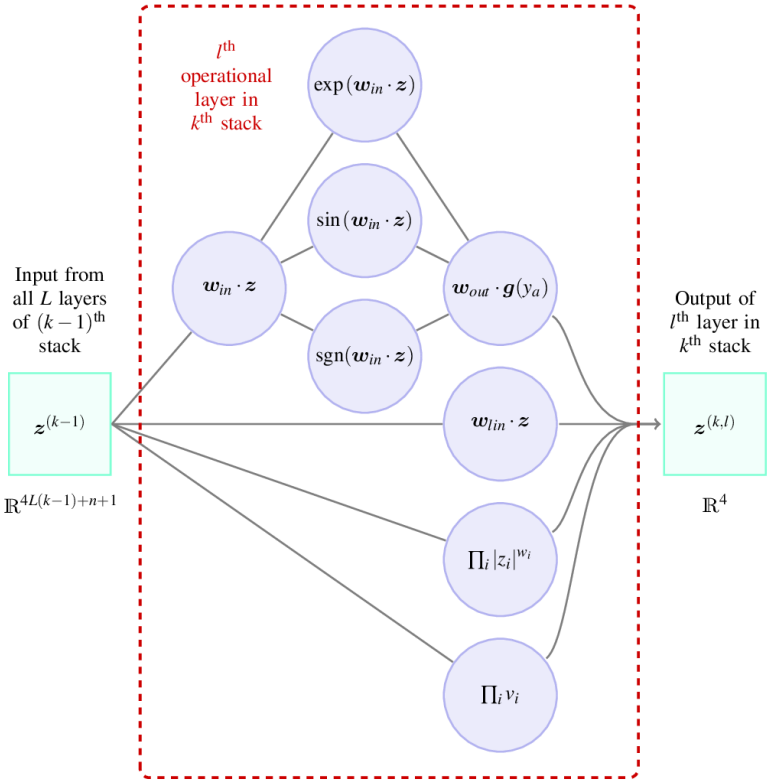

Figure 2: SymANNTEx operational layer showing (from the bottom) simple products, polynomial combinations, linear combinations, and common operators. More detail is available in [6].

2 EXAMPLES

In [6], we show that SymANNTEx can identify good models for a variety of dynamical systems from noisy data. Here we illustrate this for several examples.

2.1 Lorenz Equations

A prototypical dynamical system with chaotic dynamics is the Lorenz equations with standard parameters, originally used to model two-dimensional atmospheric convection [8]:

$$\frac{dx}{dt} = 10\,(y - x) \tag{3}$$

$$\frac{dy}{dt} = x\,(28 - z) - y \tag{4}$$

$$\frac{dz}{dt} = x\,y - \frac{8}{3}z \tag{5}$$

As input to SymANNTEx, we use 1000 data points consisting of the states and the vector field evaluated at the data points from a single randomly initialized trajectory, equally spaced in time on the time interval [0, 25]. Small noise is added to both the state and the vector field data. Using K=1 stack and L=10 layers, corresponding to 322 trainable parameters, our approach is able to learn the exact model for this system. The time series for this system is shown in Figure 3.

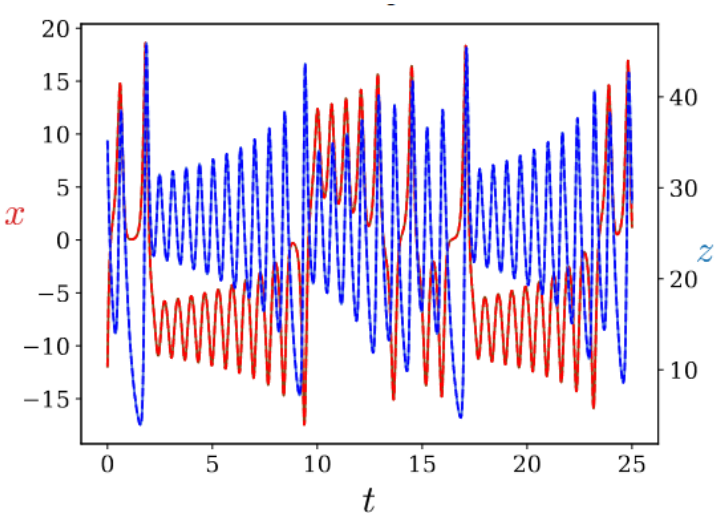

Figure 3: Time series showing x and z for the Lorenz equations (3)-(5). 1000 data points over this interval, equally spaced in time, were used to exactly identify the model using SymANNTEx.

2.2 Chemical Kinetics with Arrhenius Rate Dependence

We next consider the exponential approximation to Arrhenius rate law in chemical kinetics for two first-order reactions [9]:

$$\frac{d\alpha}{dt} = -\kappa\alpha e^{\theta} + \mu \tag{6}$$

$$\frac{d\theta}{dt} = \alpha e^{\theta} - \theta \tag{7}$$

In these equations, α is the intermediate chemical concentration, and θ represents temperature rise. For the parameters μ = 0.1 and κ = 0.07, this system has a stable limit cycle with a large separation of timescales.

As input to SymANNTEx, we use state and vector field data for a collection of short trajectories starting at random initial conditions (100 trajectories, each with 160 data points corresponding to time integration of 0.001 time units). Small noise is added to both the state and the vector field data. Using K=2 stacks and L=1 layer, corresponding to 68 trainable parameters, our approach is able to learn a model for this system which shows good agreement with the exact model; see Figure 4.

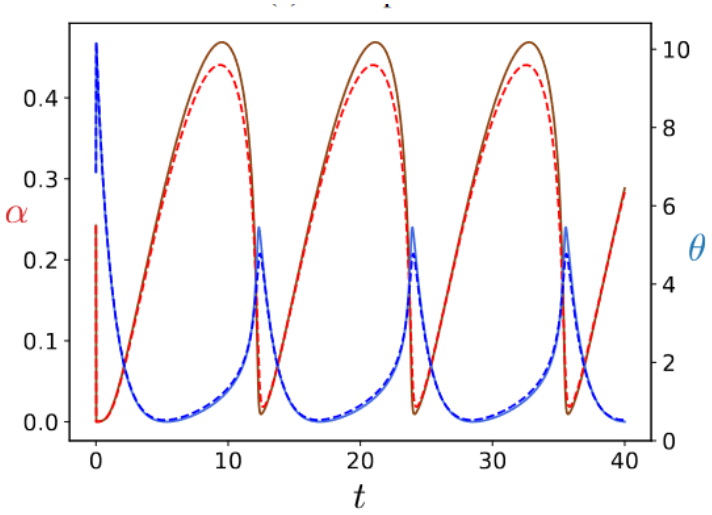

Figure 4: Time series showing α and θ for the (solid) exact model and the (dashed) approximate model learned by SymANNTEx for the chemical kinetics model (6)-(7). Although SymANNTEx does not learn the exact model, it gives a model which shows a very good approximation to the exact dynamics.

2.3 Tent Map

In addition to ordinary differential equation models, SymANNTEx can also be used to learn a map from data. Here we consider the tent map in a regime where it has chaotic dynamics:

$$x_{n+1} = 0.9\,(1 - |1 - 2x_n|) \tag{8}$$

As shown in Figure 5, SymANNTEx identifies an analytic single-humped function with a dynamic range which is similar to the tent map from 1000 data points using L = 10 layers and K = 1 stack:

$$x_{n+1} = -1.429\,sin\,(2.25\,x_n - 2.667) - 0.667 \tag{9}$$

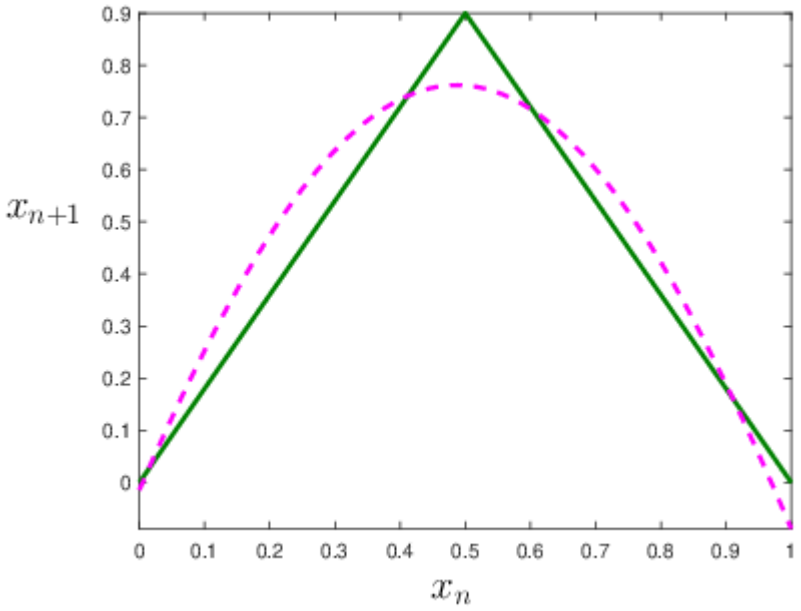

Figure 5: Identified map (9) (dashed magenta) compared to the tent map (8) (solid green).

However, the long-term dynamics for this map (9) are not a good match to the exact dynamics (8). But when we also include *abs*(·) as a primitive operation in the common operator portion of our operational layer, our approach identifies the exact model for the tent map.

3 CONCLUSIONS

SymANNTEx is a novel symbolic regression / generative dictionary method for identifying interpretable, closed-form models for a dynamical system from noisy time series and vector field data. Unlike fixed dictionary methods, it does not require pre-specification of the exact terms which form the right-hand side of the model. Rather, candidate terms are generated by combinations and compositions of the following primitive operations: linear combinations, polynomial combinations, simple products, and common operators such as $sin\,(\cdot)$, $sgn(\cdot)$, and $e^{(\cdot)}$. The depth of the network, given by the number of stacks, determines the level of complexity obtained by composing functions. The breadth of the network, given by the number of operational layers within each stack, allows multiple occurrences of a given function with the possibility of different coefficients. The effectiveness of SymANNTEx was shown here for the Lorenz equations, a chemical kinetics model, and the tent map. Additional

examples of the application of SymANNTEx are shown in [6].

We are currently considering a version of SymANNTEx which leverages ideas from neural ODEs [10] to identify models from times series data alone, without the need for vector field data. We are also exploring how SymANNTEx can be modified to learn models which obey desired symmetry properties.

Acknowledgment
This work was supported by National Science Foundation Grant No. NSF-2016004.

REFERENCES – TO BE FILLED IN

[1] Fowler, A. C. *Mathematical Models in the Applied Sciences*. Cambridge University Press, Cambridge, (1997).

[2] L. Ljung. *System Identification: Theory for the User*. Prentice Hall, Upper Saddle River, NJ, 1999. Second Edition.

[3] Bongard, J. and Lipson, H. Automated reverse engineering of nonlinear dynamical systems. *Proc. Natl. Acad. Sci. U.S.A.* (2007) **104**: 9943-9948.

[4] Quade, M., Abel, M., Shafi, K., Niven, R. K. and Noack, B. R. Prediction of dynamical systems by symbolic regression. *Phys. Rev. E* (2016) **94**: 012214.

[5] Quade, M. and Gout, J. and Abel, M. Glyph: Symbolic regression tools. *J. Open Res. Softw.* (2019) **7**: 19.

[6] Boddupalli, N., Matchen, T., and Moehlis, J. Symbolic regression via neural networks. *Chaos* (2023) **33**:083150.

[7] Akaike, H. A new look at the statistical model identification. *IEEE Trans. Autom. Control* (1974) **19**:716–723.

[8] Lorenz, E. N. Deterministic nonperiodic flow. *J. Atmos. Sci*. (1963) **20**:130–141.

[9] Gray, P. and Scott, S. K. *Chemical Oscillations and Instabilities*. Oxford University Press, Oxford, (1990).

[10] Chen, R. T. Q., Rubanova, Y., Bettencourt, J., and Duvenaud, D. K. Neural ordinary differential equations. Advances in Neural Information Processing Systems, (2018) 31:6571-6583.