A Neural Ordinary Differential Equation Network for Data-Driven Controllers

Brandon Colon, Shiqi Nan, Dr. Chunjiang Qian

The University of Texas at San Antonio, San Antonio TX, 78249

Name: *Brandon Colon*
Status: *Junior, Undergraduate Student*
Department: *Electrical and Computer Engineering*
Area of Study: *Computer Engineering*
USDA/UTSA Mentor: *Dr. Chunjiang Qian, Shiqi Nan*

WeARE Research Area

The research areas for the following work falls under data-science, non-linear control systems, and neural network. Relatable to the institute due to the goals of advancement in a neural network to model dynamic systems, which could have implications in smarter grid control. Particularly, in motor control, there is hope in advances in more precise systems that do not over correct or under correct. This advancement will allow for a more energy-efficient motor control and a higher degree of accuracy in high precision controls.

Background

Recent achievements in data-driven control system design and neural network ability to no longer be limited to a finite size had inspired this following research. The groundbreaking paper Neural Ordinary Differential Equations (ODE), originally published in 2018, is where a neural network was used to predict the behavior of ODEs in a continuously expanding network[1]. Proving for the first time, a neural network can solve and predict solutions of ODEs, and due to these advances, there is new relation to how it may be used in non-linear data-driven control of in a servo motor control[2]. Our group is continuing to work in bringing these two areas of science together for the betterment of control systems by constructing a system that takes a data-driven approach to design.

The particular neural ODE network we want to construct has advancement goals in training methods and training time, solving indifferentiable vectors fields, and practical applications such as bone modeling and control systems. Focusing on the training methods and solving indifferentiable vectors, which will be known as directions one & two, the research looks to offer solutions in solving these problems. The analytical solution methods and early hypotheses to solve these problems will be discussed in this following work.

Objectives

1. Direction one goal is to find an analytical method that could be used in a neural ODE network to expand the initial conditions or the number of inputs by using system solutions and intrinsic properties to provide training data and dynamic insights.
2. Direction two goal is to build a homogeneity inspired neural ODE, that does not have the same restraints of smoothness as the original neural ODE network using homogeneity identities.

Methodology

As for the methods to reach these directional goals, they remain theoretical, due to the current working limitations under COVID-19. Given this information, there is a good lead on how to approach and solve direction one, by using an open-source GitHub Neural ODE Demo[3]. The GitHub publisher built a neural ODE network that employs the original 2018 paper methods of solving and tracking an ODE function over time[1]. Hereby decreasing the difference between the real and estimated solutions over time[1]. Refer to figure 1 for a general visualization of an ODE neural network. Where instead of a finite number of the hidden layers, the network uses residual layers that employ a version of Euler's method to expand the number of evaluation locations to increase the accuracy of the estimated solutions. The ODE demo network could also be used to test and perfect a new method of expanding the number of inputs to decrease the models training time. Similar to the method used in the residual forward and backward Euler equations, it can be used to find an approximated solution values of the ODE in order to determine more initial conditions for the model. Refer to figure two to see the original and generalization of the forward Euler equation that could be solved using the method[4].

As for the second direction limited amount of work as been applied in order to construct a proper method of execution. The working theory is that not every ODE can be solved due to indifferentiable vector fields. The starting idea is to employ an identity of homogeneity to expand the ODE to a solvable vector field. One example of explaining this process is by using the companion system[5]. Where an ODE system of linear equations with an order higher than one can be reduced by setting the differential equal to a variable. Creating a pair of linked ODE that may be solved as a first-order ODE while preserving the homogeneity[4]. The method needs more clarification and greater consideration for first-order indifferentiable vector fields.

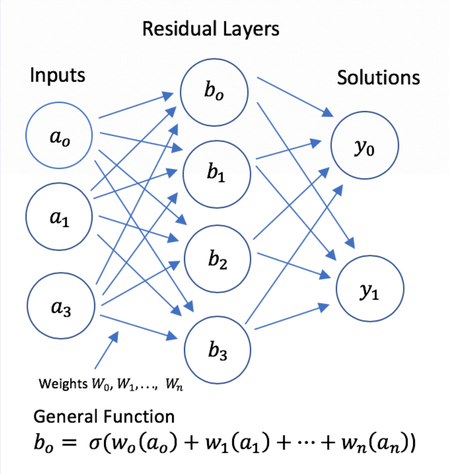

Fig. 1
Example visualization of the Neural ODE Demo

Forward Euler Method

Time-step t_n solution at nth time where $y_n \equiv y(t = t_n)$.

Step size h constant $h = t_n - t_{n-1}$ given (t_n, y_n)

$$y'(t) = \frac{y(t+h) - y(t)}{h}$$

Reduce

$$y_{n+1} = y_n + hf(t_n, y_n)$$

Approximation of nearby point

Fig. 2
Forward Euler Method

Results

As for any analytical result of the ongoing investigations, very little could be said about the achievement of the directional goals. The current method for direction one needs more testing and development before a definitive statement could be made about the precision and training speed of the new model. The method must first be proven to give a higher resolution of the predicted result when compared to the true solutions. Regardless of the current state, the goal is to have the new model with a faster training speed and an overall reduction of run time to provide a faster network.

The second direction offers a more challenging goal of finding solution methods for the indifferentiable vector fields. Proving very challenging and out of my current knowledge gap, more fundamental research is needed to find and prove solution methods for multiple cases. The direction goal is to have a more complex neural ODE network to solve all cases, even with the indifferentiable vector fields.

Skills and Experience

The project offered some of the most challenging topics I have seen in my time as an undergraduate, forcing me to take a more direct role in learning new concepts and critical thinking. I had developed many new skills thanks to this experience from learning and understanding neural networks and working on my understanding of non-linear control. Once the University closed, the project had taken a sharp turn, and I have now gained new experience in working remotely. Understanding the difficulties that come with the virtual working and how to open and remain in good contact with my mentor.

What I Learned

The project had offered a wide range of new skills to be learned and many more left to be explored due to the unique nature of the goals, which combined many areas of research in order to develop a new neural network. I had to start from the beginning with neural networks and learn the fundamentals of linear algebra associated with the neurons and solutions. This followed with the calculus used in the black box equation such as taking gradients to form a lost function and the Lambda calculus used in the ODE demo. The most valuable skills learned were working with Python, a programming language, and particularly libraries such as PyTorch and Numpy.

Future Plans

The plan is to continue in the project as long as the goals remain achievable and viable methods that still need to be considered and tested. I plan to continue on direction one over the summer and edit the neural ODE demo, with the proposed Euler method. As for the next term year, I am looking to developer a deeper understanding of the identity of homogeneity to expand ODEs. This is in order to develop a more generalized solution method for direction two, to be applied and tested.

Acknowledgments

This work is supported by the USDA National Institute of Food and Agriculture, Interdisciplinary Hands-on Research Traineeship and Extension Experiential Learning in Bioenergy/Natural Resources/Economics/Rural project, U-GREAT (Undergraduate Research, Education and Training) program (2016-67032-24984).

References

- [1] Chen, Ricky T. Q., et al. "32nd Neural Information Processing Systems." University of Toronto, Vector Institute, *Neural Ordinary Differential Equations*, 2019, pp. 1–18.
- [2] Wang, Baochao, et al. "Data-Driven Digital Direct Position Servo Control by Neural Network With Implicit Optimal Control Law Learned From Discrete Optimal Position Tracking Data." *IEEE Access*, vol. 7, 18 Sept. 2019, pp. 126962–126972., doi:10.1109/access.2019.2937993.
- [3]Chen, Ricky. "Rtqichen/Torchdiffeq/Exaples." *GitHub*, 21 Jan. 2019, github.com/rtqichen/torchdiffeq/tree/master/examples.
- [4]Zeltkevic, Michael. "Forward and Backward Euler Methods." *Forward and Backward Euler Methods*, 1998, web.mit.edu/10.001/Web/Course_Notes/Differential_Equations_Notes/node3.html.
- [5]Mattuck, Arthur, et al. "M.I.T. 18.03 Ordinary Differential Equations Notes and Exercises." *Math MIT Supplementary Notes*, M.I.T., 2017, math.mit.edu/~jorloff/suppnotes/suppnotes03/1803SupplementaryNotes_full.pdf.