Rössler Attractor Simulation and Machine Learning Visualization

Prepared by:
Nacim Osman, Tiyanna Powell
Date: October 13, 2024

Overview

This project merges chaos theory, machine learning, and user experience to simulate and predict the behavior of the chaotic Rössler attractor system. By leveraging LSTM neural networks, the tool provides a visual and interactive way to understand the system's dynamics. It also draws inspiration from my (Nacim) previous research on nonlinear systems like the Lorenz and Duffing systems and my ongoing work with multivariate physical systems.

The application aims to provide an intuitive interface for manipulating system parameters, offering users a hands-on approach to exploring complex physics. My motivation stems from a desire to combine advanced physics research with accessible machine learning tools to enhance education and exploration.

Key Features

1. **User-Friendly Interface**:
    - Adjust system parameters (a, b, c, initial conditions, etc.) via an interactive GUI.
    - Configure machine learning hyperparameters such as epochs and batch size.
    - Stop simulations mid-run for flexibility and efficiency.
2. **Integrated Machine Learning**:
    - LSTM models predict future states of chaotic trajectories.
    - Real-time Mean Squared Error (MSE) evaluation for predictions.
3. **Dynamic Visualization**:
    - **2D Plots**: Display true values, predictions, and MSE for x, y, and z.
    - **3D Plot**: Illustrates the attractor's full trajectory, combining all axes with MSE metrics.
4. **Scientific Relevance**:
    - A tool for studying nonlinear dynamics, inspired by my work on Lorenz systems, Duffing oscillators, and multivariate systems, all of which are on my GitHub.
    - Aligns with my personal research in advanced physics and chaotic systems.

How to Use

Setup

1. Install Python 3.8+.
2. Install dependencies:
   pip install numpy scipy keras scikit-learn matplotlib PyQt6

Running the Application

1. Run the script:
2. Adjust parameters and click "Run Simulation."
3. Observe progress via an animated loading bar.
4. View results in the 2D and 3D plots, complete with MSE metrics.

Stopping the Simulation

- Use the "Stop Simulation" button to cancel the process at any time.

What's Happening

- **The Rössler Attractor**:
  The system evolves based on differential equations:

$$\begin{cases} \frac{dx}{dt} = -y - z \\ \frac{dy}{dt} = x + ay \\ \frac{dz}{dt} = b + z(x - c) \end{cases}$$

  It is solved numerically using scipy's odeint.

- **Machine Learning Predictions**:
  LSTM networks train on early segments of the system's data, predicting the future

trajectory. These predictions are compared to the actual trajectory, with accuracy measured via MSE.

- **Interactive Visualization**:
  2D plots show individual axis predictions, while a 3D plot visualizes the entire attractor. MSE values highlight the prediction accuracy.

My Research Integration

This project builds on my past explorations of chaotic systems, including the Lorenz and Duffing systems. I aim to combine advanced machine learning with nonlinear physics research, offering tools that are not only technically robust but also user-friendly. By allowing users to adjust parameters, this application transforms complex concepts into approachable visualizations, enhancing both education and scientific exploration.

I continue to explore multivariate systems and advanced physics applications, utilizing tools like this to bridge theoretical research and practical insights.

Acknowledgment

I would like to thank my partner, Tiyanna Powell, for their great contributions to testing the neural network and researching advanced methods. Tiyanna also contributed to coding parts of the machine learning model. I couldn't have completed this project without her support and collaboration.

Conclusion

This project is a blend of research, machine learning, and usability. It allows users to simulate chaotic systems, experiment with machine learning predictions, and explore the beauty of nonlinear dynamics. It underscores my commitment to making advanced physics accessible through interactive and educational tools.

References

Leonov, G. A., & Kudryashova, E. V. (2020). November 1, 2024. The Lorenz system: Hidden boundary of practical stability and the difficulties of numerical analysis. *Nonlinear Dynamics*, 100(1), 631–640. https://doi.org/10.1007/s11071-020-05856-4

Riverbank Computing. (n.d.). *PyQt documentation*. November 18, 2024, from https://doc.qt.io/
Wikipedia contributors. (n.d.). *Rössler attractor*. In *Wikipedia*. October 19, 2024, from https://en.wikipedia.org/wiki/Rössler_attractor
The TensorFlow Authors. (n.d.). *Keras API reference*. In TensorFlow Core v2.0. November 6, 2024, from https://www.tensorflow.org/api_docs/python/tf/keras
Scikit-learn developers. (n.d.). *scikit-learn: Machine learning in Python*. November 3, 2024, from https://scikit-learn.org/
Deepak Venugopal. (2024). *Class presentations, slides, and assignments for Machine Learning*. August 26, 2024 University of Memphis.