VJSTI Vol. 1, 2024 pp. 1-12

Visayan Journal of Science, Technology, and Innovation

Impact of neuron count and learning rate on the accuracy of a NumPy-based artificial neural network in predictive modeling of nonlinear dynamical systems

David Mark M. Mueller

School of Engineering, Computer Studies, and Architecture (SECSA), Southland College, Kabankalan City, Negros Occidental

ABSTRACT

The impact of neuron count and learning rate on the accuracy of an artificial neural network (ANN) in predictive modeling of nonlinear dynamical systems is explored. This study is focused on the Lorenz, Rössler, and Chen systems, which are renowned for their sensitivity to initial conditions, intricate dynamics, and strangely attractive plot of their trajectories. The model's performance was assessed using symmetric mean absolute percentage error (SMAPE) and coefficient of determination (R^2). The results shows that the model with 24 neurons and 0.1 learning rate consistently outperformed other parameters across all three systems.

KEYWORDS

chaotic system, forecasting, machine learning, complex dynamics

Received: 07 July 2023
Received in revised form: 05 August 2023
Accepted: 01 September 2023
Available online: 05 December 2024

INTRODUCTION

Dynamical systems can be found in a multitude of disciplines, including weather forecasting, social dynamics, and finance. Traditional modelling and analysis methods often found themselves inadequate in capturing the nonlinear and intricate nature of such systems. Researchers have grappled with the inherent complexity of nonlinear dynamical systems, which are characterized by their sensitivity to initial conditions and their unpredictable progression over time.

A common approach to analyzing these systems is through time series analysis. However, traditional time series analysis methods have limitations when dealing with nonlinear data, mainly because they are based on linear assumptions and cannot capture the complex dynamics of nonlinear systems (Kantz & Schreiber, 2004). This has led to misinterpretations and inaccurate forecasts when using such methods for predicting nonlinear and chaotic systems.

The pioneering efforts by Narendra and Parthasarathy to utilize ANNs for predictive modelling and control of chaotic systems marked the beginning of an exciting exploration of this area in the early 1990s (Narendra & Parthasarathy 1990; Narendra & Parthasarathy 1992).

This study aims to explore the impact of neuron count and learning rate on the accuracy of a numpy-based neural network in predictive modeling three-dimensional nonlinear dynamical systems. In contrast to existing neural network libraries employed for predictive modeling, libraries such as TensorFlow, Scikit-learn and Keras (Wagh, 2020), this study takes a less common approach where the neural network used was developed entirely using only the NumPy library.

*Corresponding Author:
David Mark M. Mueller; davidmarkmueller@gmail.com

MATERIALS AND METHODS

Lorenz system

The Lorenz system is renowned for giving rise to a chaotic attractor, a sophisticated geometric depiction of the progression of the system over a given time interval. This attractor is distinguished by its complex behavior and a unique characteristic of sensitivity to its initial conditions. Minuscule variations in the starting conditions could give rise to significantly divergent outcomes (Lorenz, 1963). The Lorenz system is defined by the following equations:

$$\frac{dx}{dt} = \sigma(y - x),$$
$$\frac{dy}{dt} = x(\rho - z) - y,$$
$$\frac{dz}{dt} = (xy - \beta z), \qquad (1)$$

where σ, ρ, and β are parameters that control the behavior of the system and x, y, and z are the state variables of the system.

Rössler system

The Rössler system, proposed by Otto Rössler in 1976, is another example of a three-dimensional chaotic attractor that possesses a spiral structure. The system's trajectory wraps around this attractor, following a complex pattern that enhances its unpredictability (Rössler, 1976). The mathematical formulation of the Rössler system is defined by the following equations:

$$\frac{dx}{dt} = -y - z,$$
$$\frac{dy}{dt} = -x + ay, \qquad (2)$$
$$\frac{dz}{dt} = b + z(x - c)$$

where x, y, and z are the state variables, and a, b, and c are system parameters and x, y, and z are the state variables of the system.

Chen system

Another renowned chaotic system that stands out for its distinctive behavior and structure (Chen, 1999). The mathematical formulation of the Chen system is defined by the following equations:

$$\frac{dx}{dt} = a(y - x),$$
$$\frac{dy}{dt} = (c - a)x - xz + cy,$$
$$\frac{dz}{dt} = xy - bz \qquad (3)$$

where x, y, and z are the state variables, and a, b, and c are system parameters and x, y, and z are the state variables of the system.

Data Generation

For each system 10,000 observations were generated. The observations were then used as training and testing datasets for the neural network model. The parameters and initial values used for each system are as follows:

i. Lorenz ($\sigma = 10,\ \beta = \frac{8}{3},\ \rho = 35;\ x_o = 1,\ y_o = 1,\ z_o = 1$)
ii. Rossler ($a = 0.2,\ b = 0.2,\ c = 5.7;\ x_o = 0.1,\ y_o = 0.1,\ z_o = 0.1$)
iii. Chen ($a = 35,\ b = 3,\ c = 28;\ x_o = 1,\ y_o = 1,\ z_o = 1$)

Performance metrics

To assess the performance of the machine learning model SMAPE and R^2 were used.
SMAPE, an enhanced version of MAPE (mean absolute percentage error), measures the relative error between the predicted and the actual values. It falls within the range of (0%, +∞), wherein lower values signify enhanced predictive accuracy. An ideal performance yields a SMAPE of 0%, signifying a flawless predictive model (Jierula et al, 2021). It is calculated using the equation (4):

$$SMAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_{pre} - y_{act}}{(|y_{pre}| - |y_{act}|)/2}\right| \quad (4)$$

The R^2 metric are used to measure how well observed outcomes are replicated by the model.
The R^2 metric are used to measure how well observed outcomes are replicated by the model. It measures the proportion of the variance in the dependent variable that is explained by the independent variable. The formula for R^2 are calculated using the equations (5):

$$R^2 = 1 - \frac{\sum_1^n (y_{act} - y_{pre})^2}{\sum_1^n (y_{act} - \overline{y_{act}})^2} \quad (5)$$

where $\overline{y_{act}}$, is the mean of the actual value, and $\overline{y_{pre}}$ is the mean of the predicted value.

Artificial neural networks

ANNs have garnered recognition for its effective performance in predictive modelling tasks such as classification and regression, often outdoing traditional statistical models (LeCun et al., 2015). It has the potential to model complex, nonlinear relationships between input and output variables, making it an ideal candidate for addressing the limitations of traditional time series analysis methods in predicting nonlinear data.

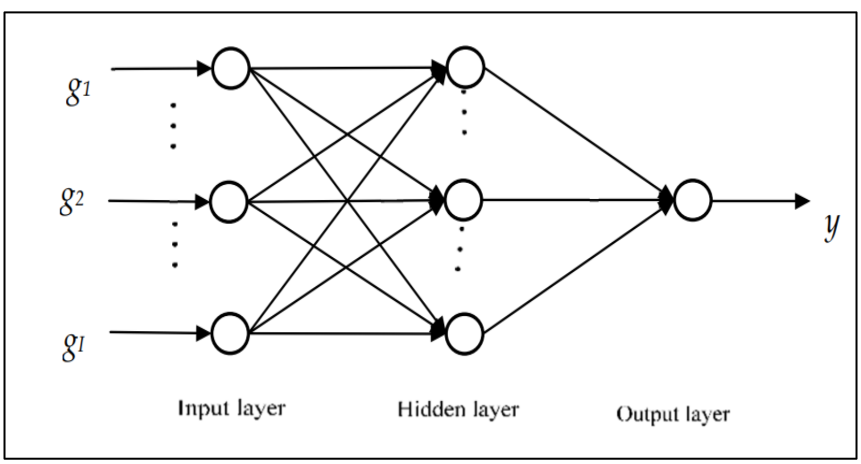

Figure 1. Schematic diagram of an artificial neural network with three layers (Isabona et al, 2022)

Figure 1 illustrates a standard neural network model. The network structure is comprised of an input layer (g_1, g_2, ..., g_i), a predicted output (y_1, y_2,, ..., y_h), and k_h hidden nodes. Each layer is differentiated by a series of weighted connections, marked as w_{ij}^1 and w_{jn}^2 for the links between the input-hidden and hidden-output layers, respectively. The thresholds for the hidden nodes are denoted by c_j.

The output from the neural network, specifically corresponding to the jth neuron linked with the kth node, can be expressed using Equation 7.

$$\hat{y}_n(t) = \sum_{j=1}^{k_h} w_j^2 F\left(w_{ij}^1 g_i(t) + c_j\right) \qquad (7)$$

$$for\ 1 \leq n < m, 1 \leq j < k_h, (w_j, j = 0, 1, \ldots, k_h), (w_{ij}, i = 0, 1, \ldots, m; j = 0, 1, \ldots, k_h)$$

where m, h, and k_h indicate the input node number, hidden node, and hidden node number, respectively, and i represents the input to the jth hidden layer neuron. The sigmoid activation function was used in this study and is defined by Equation 8,

$$F(a) = \frac{1}{1 + e^{-a}} \qquad (8)$$

Where *F(a)* being a set of real numbers (Isabona et al, 2022).

The algorithm of the ANN implemented in this study is shown in Figure 2.

Algorithm 1 Artificial neural network

1. **Define Functions**
   - Sigmoid(x) ⇒ return $\frac{1}{1+e^{-x}}$
   - ForwardProp(X, W1, b1, W2, b2)
     - Compute Z1 = X*W1 + b1
     - Apply Sigmoid activation function: A1 = Sigmoid(Z1)
     - Compute Z2 = A1*W2 + b2
     - return Z2, A1
   - BackProp(X, y, $\hat{y}$, A1, W2)
     - Compute dZ2 = $\hat{y}$- y, dW2 = (1/m) * A1.T * dZ2
     - Compute db2 = (1/m) * sum(dZ2), dZ1 = dZ2 * W2.T * A1 * (1 - A1)
     - Compute dW1 = (1/m) * X.T * dZ1, db1 = (1/m) * sum(dZ1)
     - return dW1, db1, dW2, db2
2. **Load and split data** into training and testing sets
3. **Initialize weights** and biases
4. **Training**
   - For each epoch do:
     - ForwardProp(X, W1, b1, W2, b2)
     - BackProp(X, y, $\hat{y}$, A1, W2)
     - Update weights and biases: W1 -= lr * dW1, b1 -= lr * db1, W2 -= lr * dW2, b2 -= lr * db2
5. **Evaluate model**
   - Calculate and print evaluation metrics
6. **Export predictions** to a CSV file
7. **Visualize** original and predicted data

Figure 2. Algorithm of the ANN model employed

RESULTS

Table 1. Accuracy ratings of Lorenz system at 10,000 observations and 1000 Epochs.

3 neurons	***SMAPE***	R^2
Learning Rate		
0.001	0.06700	0.06800
0.01	0.06600	0.00600
0.1	0.05400	0.29200
6 neurons	***SMAPE***	R^2
Learning Rate		
0.001	0.06610	0.02950
0.01	0.03930	0.60510
0.1	0.02980	0.81010
12 neurons	***SMAPE***	R^2
Learning Rate		
0.001	0.06350	0.05070
0.01	0.02720	0.82110
0.1	0.02910	0.82140
24 neurons	***SMAPE***	R^2
Learning Rate		
0.001	0.05920	0.16840
0.01	0.02456	0.86050
0.1	0.02740	0.84120
36 neurons	***SMAPE***	R^2
Learning Rate		
0.001	0.06047	0.13550
0.01	0.02440	0.87105
0.1	0.02545	0.86123

Table 2. Accuracy ratings of Rössler system at 10,000 observations and 1000 Epochs.

3 neurons	***SMAPE***	R^2
Learning Rate		
0.001	0.43468	0.03813
0.01	0.40418	0.17765
0.1	0.37604	0.29339
6 neurons	***SMAPE***	R^2
Learning Rate		
0.001	0.39174	0.09357
0.01	0.39608	0.18399
0.1	0.30781	0.33243
12 neurons	***SMAPE***	R^2
Learning Rate		
0.001	0.39705	0.13582
0.01	0.40060	0.19649
0.1	0.35773	0.32637

24 neurons	***SMAPE***	R^2
Learning Rate		
0.001	0.38937	0.17006
0.01	0.39414	0.23619
0.1	0.32590	0.34389
36 neurons	***SMAPE***	R^2
Learning Rate		
0.001	0.39987	0.18369
0.01	0.38970	0.23968
0.1	0.33862	0.34583

Table 3. Accuracy ratings of Chen system at 10,000 observations and 1000 Epochs.

3 neurons	***SMAPE***	R^2
Learning Rate		
0.001	0.06250	-0.12865
0.01	0.05635	0.03188
0.1	0.03822	0.51097
6 neurons	***SMAPE***	R^2
Learning Rate		
0.001	0.05859	-0.03246
0.01	0.04633	0.28140
0.1	0.04056	0.48095
12 neurons	***SMAPE***	R^2
Learning Rate		
0.001	0.05780	-0.01999
0.01	0.03172	0.62217
0.1	0.03192	0.64320
24 neurons	***SMAPE***	R^2
Learning Rate		
0.001	0.05588	0.04042
0.01	0.03292	0.60669
0.1	0.02895	0.68775
36 neurons	***SMAPE***	R^2
Learning Rate		
0.001	0.05591	0.03474
0.01	0.03026	0.65069
0.1	0.03098	0.65400

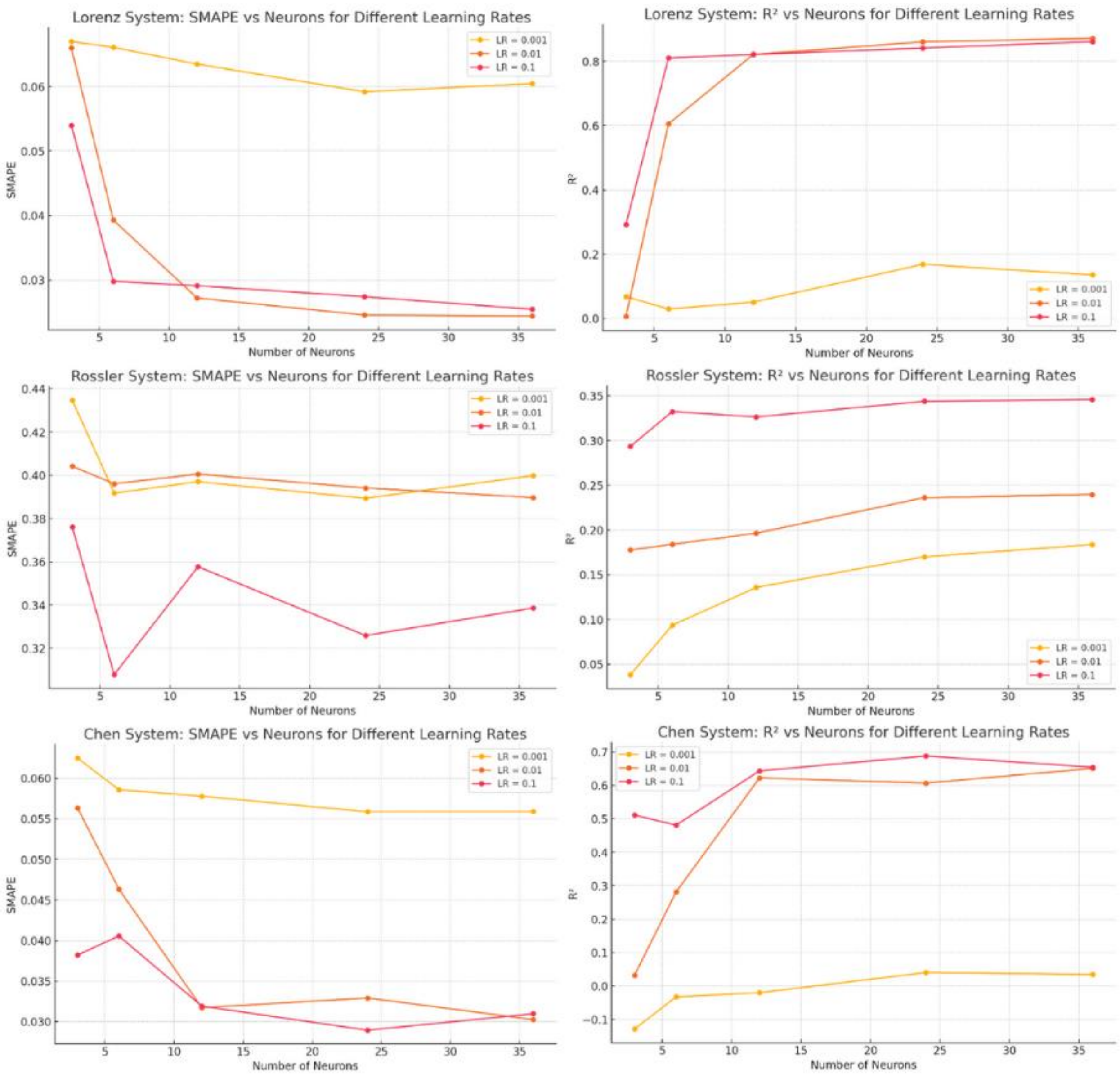

Figure 3. Comparison of model accuracy metrics for Lorenz, Rössler, and Chen Systems.

The presented results are the performance of the ANN models trained to forecast the dynamics of the Lorenz, Rössler, and Chen systems. Five separate models were trained, each with varying neurons: 3, 6, 12, 24, and 36 in each layer. Furthermore, learning rates of 0.001, 0.01, and 0.1 were employed for training these models.

An analysis of the Lorenz system reveals a positive correlation between the number of neurons in the neural network and the performance of the model. It is evident that the model equipped with 36 neurons outperforms others with the lowest SMAPE and highest R^2 values. This indicates superior accuracy in predicting the Lorenz system compared to the models with 3, 6, 12, and 36 neurons. As for the learning rate, a value of 0.01 consistently led to the optimal performance for the Lorenz system.

The Rössler system mirrors a similar trend. As the number of neurons escalates, so does the performance of the model. The model with 24 neurons exhibits the lowest SMAPE scores and the highest R^2 values, reiterating its accuracy in predicting the Rössler system. As observed in the Lorenz system, the model with 3 neurons underperforms, and the model with 6 neurons cannot rival the accuracy of the models with 12, 24, and 36 neurons. A learning rate of 0.1 ensures the best performance for the Rössler system (Table 2).

An increase in the number of neurons enhances the performance of the Chen system model. The model with 3 neurons lags behind, while the one with 24 neurons leads, much like in the other systems. A learning rate of 0.1 proves optimal for the Chen system too (Table 3).

These results suggest that increasing the neuron count in the neural network bolsters performance when predicting these chaotic systems. Moreover, a learning rate of 0.1 emerges as the most effective across all three systems.

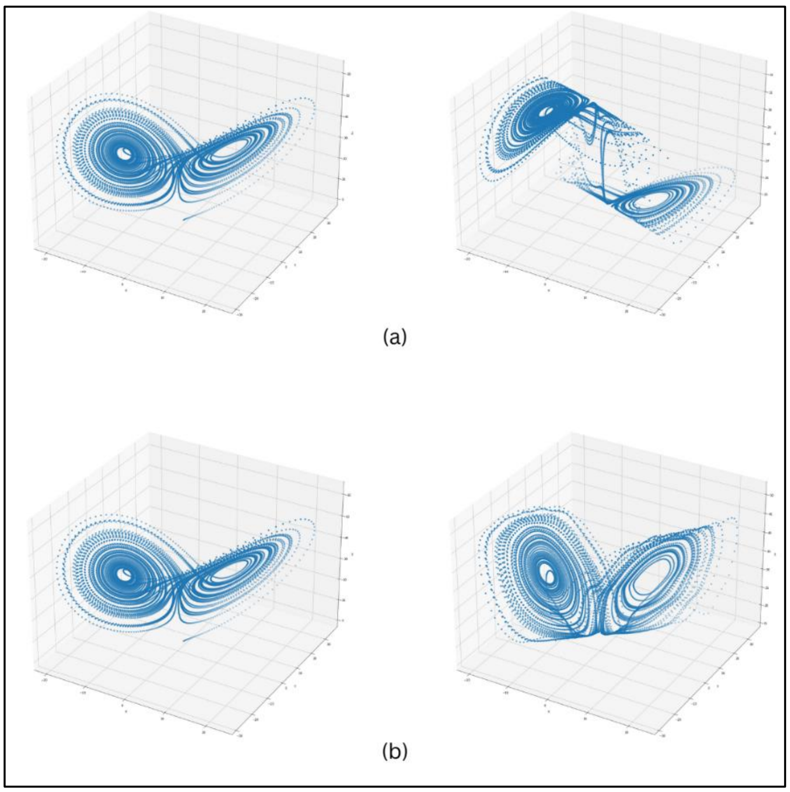

Figure 3. (a) Lorenz system original data vs predicted data at 3 neurons and 0.001 learning rate, (b) Lorenz system original data vs predicted data at 12 neurons and 0.01 learning rate

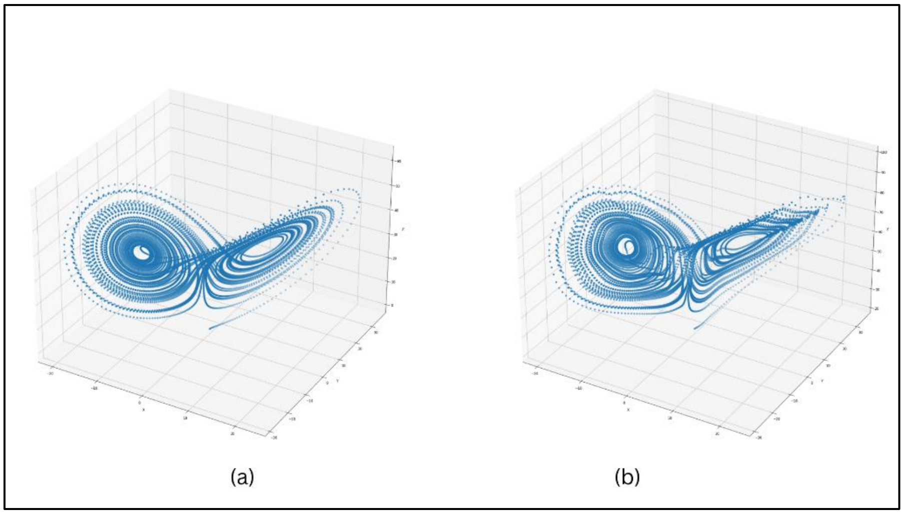

Figure 4. (a) Lorenz system original data, (b) predicted data at 36 neurons and 0.01 learning rate

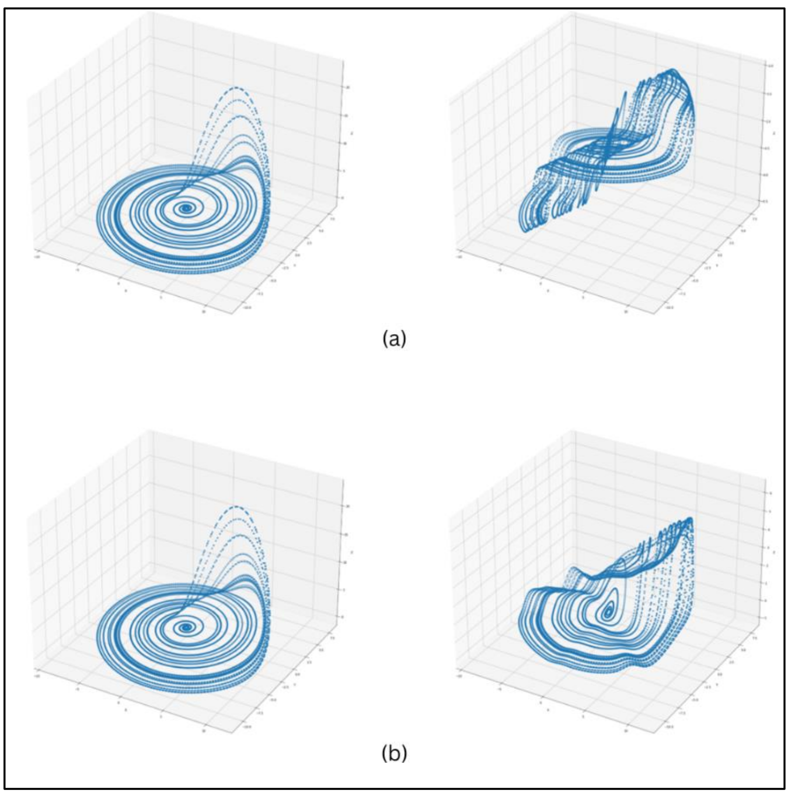

Figure 5. (a) Rössler system original data vs predicted data at 3 neurons and 0.001 learning rate, (b) Rössler system original data vs predicted data at 12 neurons and 0.01 learning rate

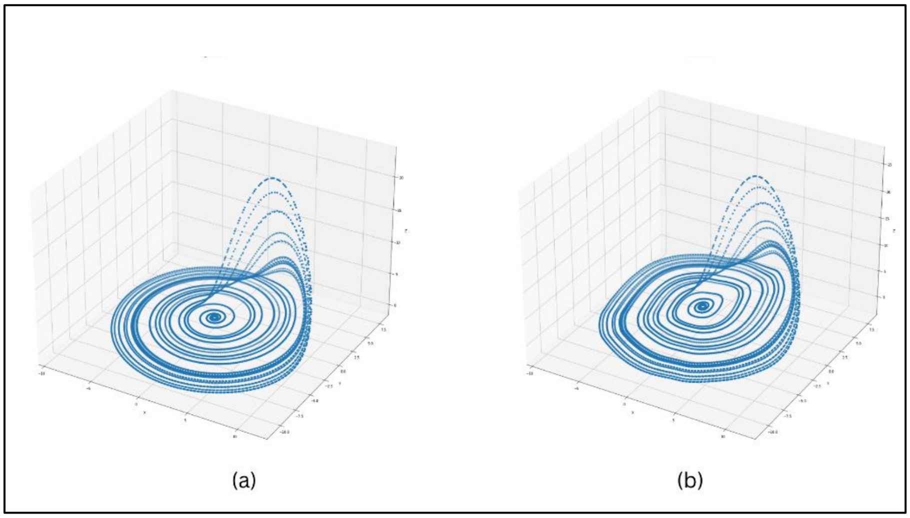

Figure 6. (a) Rössler system original data, (b) predicted data at 24 neurons and 0.1 learning rate

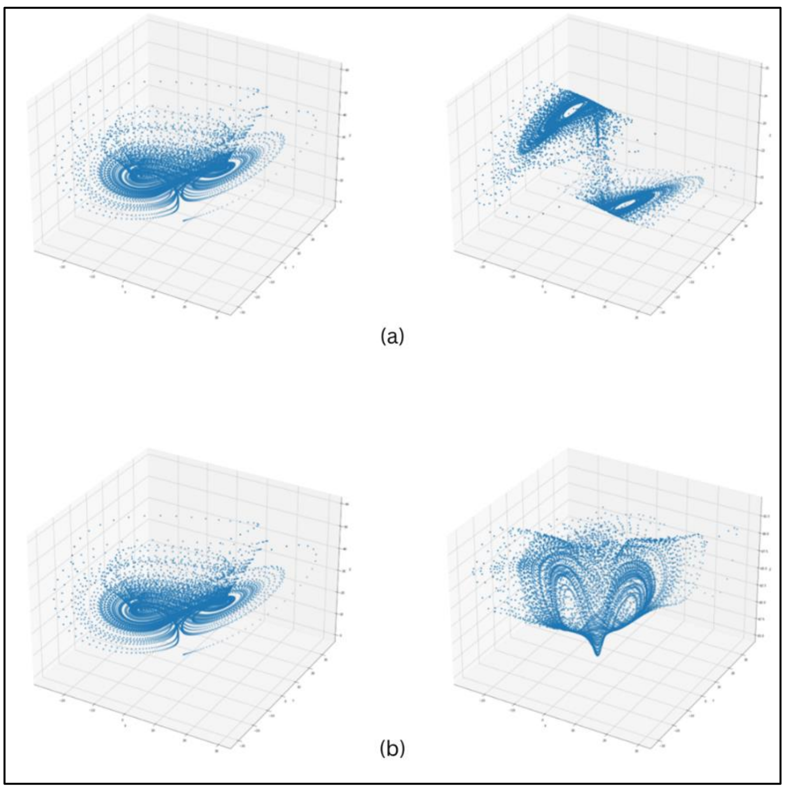

Figure 7. (a) Chen system original data vs predicted data at 3 neurons and 0.001 learning rate, (b) Chen system original data vs predicted data at 12 neurons and 0.01 learning rate

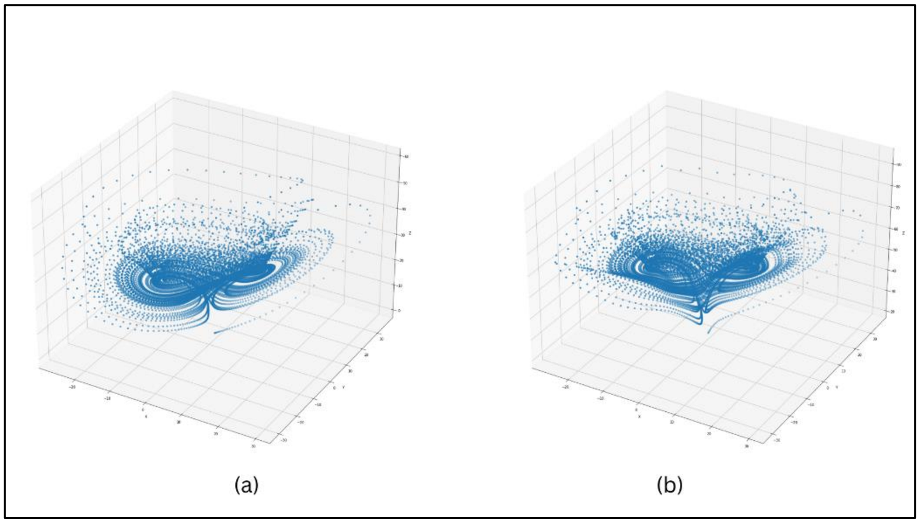

Figure 8. (a) Chen system original data, (b) predicted data at 24 neurons and 0.1 learning rate

DISCUSSION

The objective of this study was to assess the impact of neuron count and learning rate on the accuracy of a NumPy-based ANN in predicting three well-known nonlinear dynamical systems: Lorenz, Rössler, and Chen. The models were trained on various combinations of neurons (3, 6, 12, 24, and 36) and learning rates (0.001, 0.01, and 0.1) to investigate their influence on the predictive accuracy. In summary, the overall best-performing models were as follows:

Lorenz System: 36 neurons with a learning rate of 0.01.
Rössler System: 24 neurons with a learning rate of 0.1.
Chen System: 24 neurons with a learning rate of 0.1.

Findings reveal that the model endowed with 24 neurons and 0.1 learning rate consistently outperformed others across all three systems.

Careful selection of the learning rate and number of neurons are suggested as these factors can notably enhance the predictive accuracy of the neural network when dealing with chaotic systems. Further investigations focusing on additional factors like the number of epochs, the choice of activation function, and the specific architecture of the neural network are recommended.

ACKNOWLEDGEMENT

Sincere gratitude is extended to Southland College and Central Philippines State University for their invaluable support. Special thanks to Dr. Christopher Caipang for his guidance and expertise in editing this paper, and to Ms. Tabitha Famatid for her kind assistance throughout this project.

ETHICAL DECLARATION

All research has been conducted in accordance with the ethical principles and guidelines.

CONFLICT OF INTEREST

The author declares no conflict of interest.

REFERENCES:

Kantz, H., & Schreiber, T. (2004). Nonlinear time series analysis. Cambridge University Press.

Narendra, K. S., & Parthasarathy, K. (1990). Identification and control of dynamical systems using neural networks. IEEE Transactions on Neural Networks, 1(1), 4-27.

Narendra, K. S., & Parthasarathy, K. (1992). Neural networks and dynamical systems. International Journal of Approximate Reasoning, 6, 109-131.

Wagh, A. (2020). *Deep Learning of Nonlinear Dynamical System*, Michigan Technological University. Digital Commons @ Michigan Tech. https://digitalcommons.mtu.edu/cgi/viewcontent.cgi?article=2180&context=etdr

Lorenz, E. N. (1963). Deterministic nonperiodic flow. Journal of the Atmospheric Sciences, 20(2), 130-141.

Rössler, O. E. (1976). An equation for continuous chaos. Physics Letters A, 57(5), 397-398.

Chen, G. (1999). Controlling chaos and bifurcations in engineering systems. CRC Press.

Jierula, A., Wang, S., Oh, T.-M., & Wang, P. (2021). Study on accuracy metrics for evaluating the predictions of damage locations in deep piles using artificial neural networks with acoustic emission data. Applied Sciences, 11(5), 2314.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436-444.

Isabona, J., Imoize, A. L., Karunwi, O., Kim, Y., Lee, C.-C., Li, C.-T., & Ojo, S. (2022). Development of a Multilayer Perceptron Neural Network for Optimal Predictive Modeling in Urban Microcellular Radio Environments. Applied Sciences, 12(11), 5713.