Sheil Kumar
sk17@illinois.edu
2179043997

Learning Dynamic Systems

2021-03-16

1 Introduction

2 Data

2.1 Data Generation

We generate data from the Lorentz attractor defined by the equations:

$$\frac{dx}{dt} = \sigma(y - x) \tag{1}$$
$$\frac{dy}{dt} = x(\rho - z) - y \tag{2}$$
$$\frac{dz}{dt} = xy - \beta z \tag{3}$$

Where $\sigma, \rho,$ and β are constants.To generate the data, we make use of the Forward Euler Integration Scheme which works recursively by solving

$$y'(t) = f(t, y(t)) \tag{4}$$
$$y(t_0) = y_0$$
$$t_n = t_0 + nh$$
$$\therefore y_{n+1} = y_n + hf(t_n, y_n) \tag{5}$$

Where h is the **step size** and t_i is the i^{th} **time step**. Identical equations exist for both x and z as well. We begin with the initial value $(x_0, y_0, z_0) = (0, 0, 0)$ and solve recursively as defined above to obtain $\mathbf{X} = X_0, X_1, ..., X_n$, where $X_i = (x_i, y_i, z_i)$ for $n = 16000$.

2.2 Training and Testing

Let us define the target vector, $\mathbf{Y} = Y_0, Y_1, ..Y_{n-1}$, where $Y_i = (x_{i+1} - x_i, y_{i+1} - y_i, z_{i+1} - z_i) = X_{i+1} - X_i$. The target vector $\mathbf{Y}$ is the vector we will **train the model to predict**. As such the model will be fed the true $\mathbf{X}$ and true $\mathbf{Y}$ and be asked to predict a $\hat{Y}_i$ for each time step. That is, each X_i will be mapped to a $\hat{Y}_i$ as predicted by the model; $X_i \longmapsto \hat{Y}_i$. The $\hat{Y}_i$ determined by the model are predicted based on a set of weights $\mathbf{W}, \mathbf{U}$ and a set of bias terms $\mathbf{B}$ which are determined by the model while training to minimize the loss function, $\mathbf{L} = (\mathbf{Y} - \hat{\mathbf{Y}})^2$, this loss function is simply the **mean squared error**, where $\hat{\mathbf{Y}} = \hat{Y}_0, \hat{Y}_1, ..., \hat{Y}_{n-1}$. The specific method through which the weights, $\mathbf{W}, \mathbf{U}$ and bias terms, $\mathbf{B}$ are determined is not imperative to understand, and these will change depending on the size and architecture of the model, however **the model itself** will be discussed to some extent in the following section.

3 Model

3.1 Network Structure

We have analyzed the Lorentz attractor through many different neural networks. However, the structure of the neural networks remains the same throughout and they all recieved the same data as described in **2.1**. Each neural network is first composed of an LSTM layer composed of a certain number of LSTM units, this is then fed into a dense layer composed of three units, respectively the (x, y, z) components of the attractor, that is then returned to the user by the network. We evaluated networks LSTM units ranging between 8 and 256 units, while the number of epochs were also varied.

4 Results

Each network was evaluated according to how well it was able to reproduce the literature Lyapunov exponent for it's given attractor. The networks were trained with data pertaining to a Lorentz attractor with $\beta = \mathbf{8/2}, \rho = \mathbf{28}, \sigma = \mathbf{10}$, the literature value of the Lyapunov exponent corresponding to these parameters is $\mathbf{0.9056}$.

4.1 Lyapunov Exponents

The Lyapunov exponents for the various LSTM's were as follows.

4.1.1 256 LSTM Units

The network with 256 LSTM units was only tested under 25 epochs and it performed well.

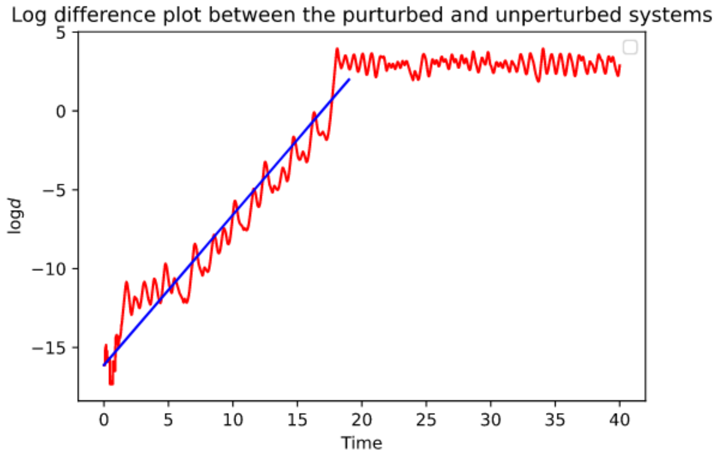


Figure 1: Lyapunov Plot for Network with 256 LSTM units

As can be seen the Lyapunov exponent, **blue line**, shares a similar gradient to the plot, thus we can evaluate this network to be succesful.

4.1.2 128 LSTM units

The network with 128 LSTM units performs similarly to the network with 256 units.

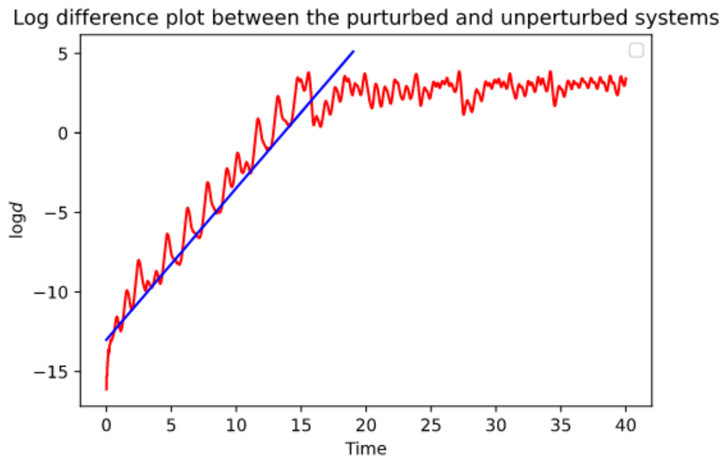


Figure 2: Lyapunov Plot for Network with 128 LSTM units

4.1.3 64 LSTM units

We are surprised by the results from the network with 64 LSTM units, as it does not appear to predict the Lyapunov exponent as well as the previous networks or the networks with even few LSTM units as can be seen in **3.1.4**

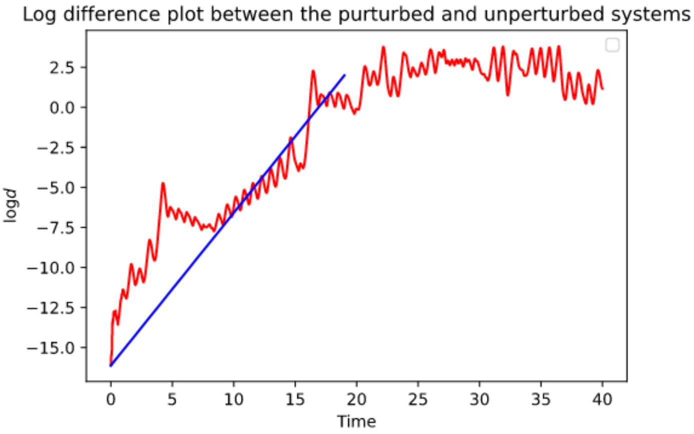
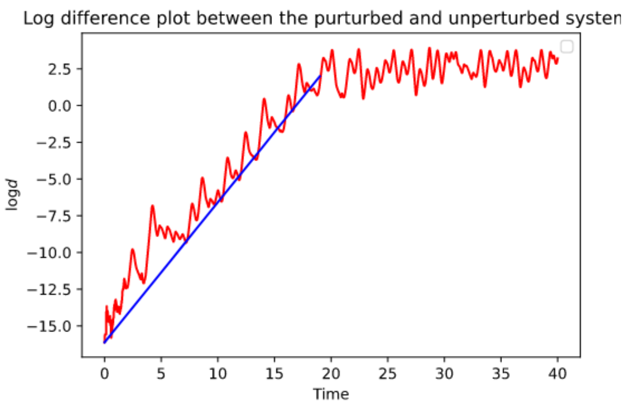


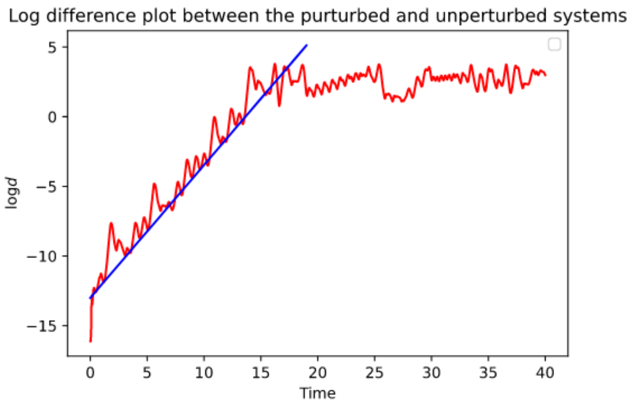
Figure 3: Lyapunov Plot for Network with 64 LSTM units

4.1.4 32 LSTM units

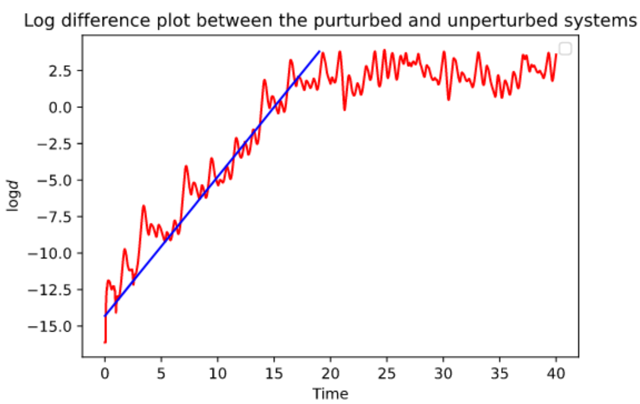
The network with 32 LSTM units were tested for 25 epochs as with the previous networks, but the effects of training for 20 and 15 networks was also evaluated.



(a) 25 Epochs



(b) 20 Epochs



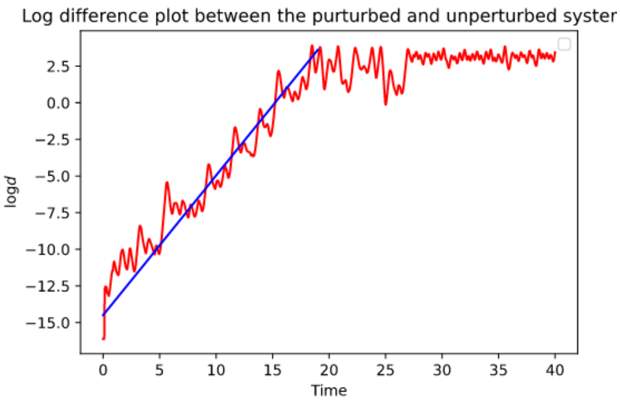
(c) 15 Epochs

Figure 4: Lyapunov Plots for the Networks with 32 LSTM units

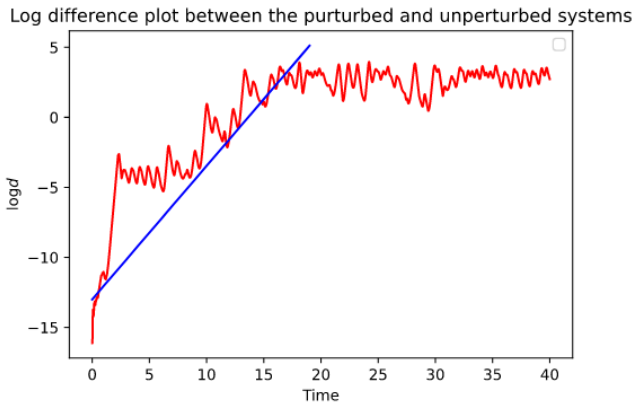
All the networks using 32 LSTM units seem to perform quite well, as they reproduce the literature value for the Lyapunov exponent quite well.

4.1.5 16 LSTM units

The networks with 16 LSTM units finally show breakdown, while the network trained for 25 epochs still reproduces the literature Lyapunov exponent, the network trained for 20 epochs breaks down.



(a) 25 Epochs



(b) 20 Epochs

Figure 5: Lyapunov Plots for the Networks with 16 LSTM units

4.1.6 8 LSTM units

The system with 8 LSTM units finally breaks down completely when trained for 25 epochs

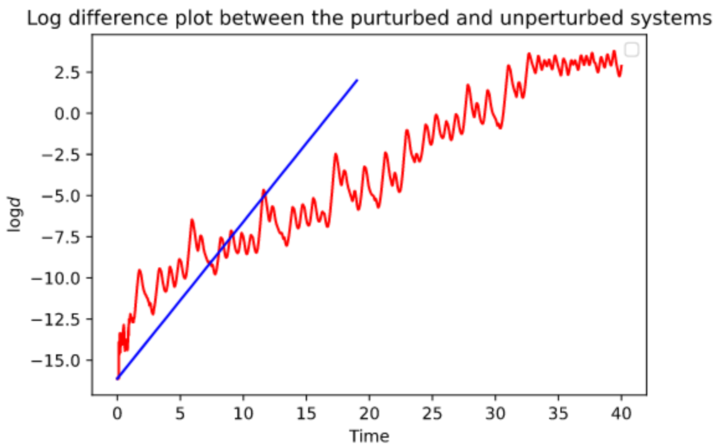


Figure 6: Lyapunov Plot for Network with 8 LSTM units