Preliminary draft

23 March 2024
brian.gitahi@alleninstitute.org

CEBRA: Performance Analysis

Brian Gitahi, Alex Piet, Uygar Sumbul
AIND, 615 Westlake Avenue Seattle, WA

Keywords: CEBRA, Embeddings, dimension reduction, dimensionality, consitent

Summary

The goal of my project is a cloud-based single-trial analysis platform for large-scale physiological recordings. Up until this point, this has taken the form of researching and deploying an efficient method to use CEBRA on Allen Institute data-sets that can be used by anyone working at the institute now and in the future. This has been done through the online platform, Code Ocean, where I am working with others in Neural Dynamics to create a pipeline to process Fiber photometry data. One of the possible analyses that will be performed on the fiber photometry data (after processing) is deploying manifold embedding techniques to find lower dimensional representations of neuromodulator dynamics. This summary documents part of my work analysing the usefulness of CEBRA to this end. We first start by analysing its performance on simple, known inputs.

Contents

1 Introduction

1.1 What is CEBRA?

"CEBRA is a library for estimating Consistent EmBeddings of high-dimensional Recordings using Auxiliary variables. It contains self-supervised learning algorithms implemented in PyTorch, and has support for a variety of different datasets common in biology and neuroscience." [1]

1.2 How does it work?

It takes time-series data and outputs a lower-dimensional embedding with latent variables that allows scientists to understand the structure of their data. It is typically used with neural data and optional auxiliary variables as input. This is one of it's distinct features that makes it a milestone for neuroscience research – that it can combine neural and behavioural data to produce lower dimensional representations of neural dynamics. Another unique feature of CEBRA is that it should give consistent embeddings across repeated runs of the algorithm or across different sessions.

CEBRA uses a contrastive learning approach to identify similar and dissimilar datapoints. It also makes use of the InfoNCE loss [1], which is a type of contrastive loss function.

2 CEBRA Performance Analysis

This section will take a closer look at the embeddings CEBRA produces, starting with simple synthetic data whose structure is well known. Specifically, we will be analysing embeddings of a simple 2D circle and the Lorenz attractor. We expect that CEBRA will be able to represent these two inputs very well in 3 dimensions, so the first step is to find what model parameters would be best for these inputs.

2.1 Dimensionality Analysis

The first thing we did was to check whether CEBRA produces reasonable embeddings for simple known inputs like a 2D circle and the Lorenz attractor. So we first played around with the output dimension parameter to see if the best embeddings were produced at the intrinsic dimensions of the inputs. Here, we used the R^2 metric to quantify the quality of the embeddings produced - a higher R^2 score means a high quality embedding was produced. We also wanted to verify that the embeddings retain the same structure in all dimensions.

First, we found that for specific model-architectures, CEBRA constrains all embeddings to the surface of a sphere in 3D or the circumference of a circle in 2D [2]. Circles and ellipses of different radii were represented by the same circle of radius 1 in 2D. We then added some noise to one of the circles and compared the R^2 scores for the original circle and the noisy circle. What we found was that the best R^2 for both the noisy and original circle was at an

[1] NCE stands for Noise-Contrastive Estimation

[2] All models contrain the dynamics of the lorenz attractor to the circumference of a circle when the output dimesion is set to 2

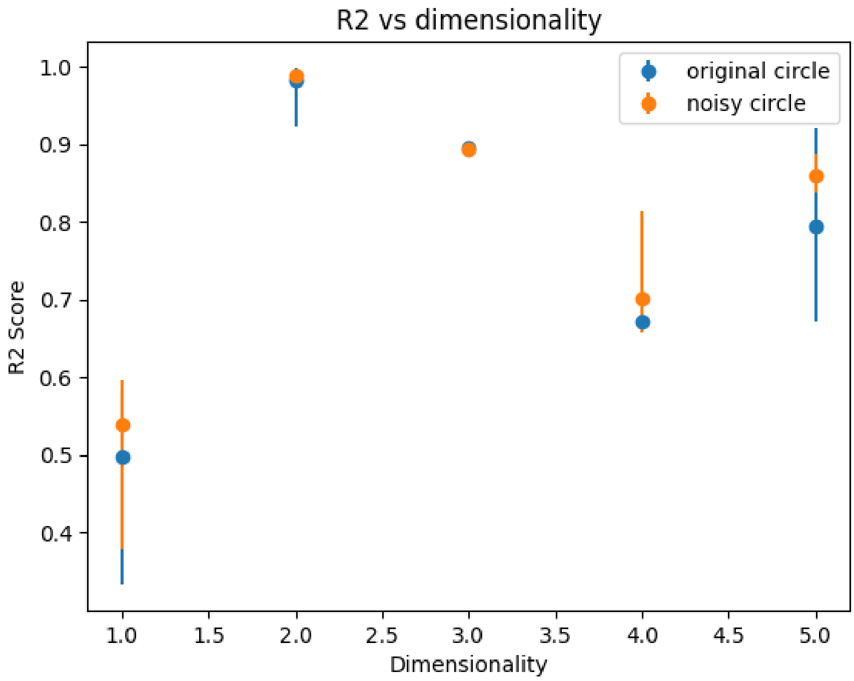

Figure 1: Figure showing the R^2 of the embeddings produced at different output dimensions for the two circles.

output dimension of 2 (see figure 1). This means that CEBRA was able to recover the input structure best in 2 dimensions – which is what we expected.

2.2 Lorenz Attractor Analysis

We then did the same thing for the Lorenz attractor, and found that the best integer dimensionality was 2 (see figure 2). [3]

However, when we use an output dimension of 2 with CEBRA, it still tries to constrain the dynamics to the surface of a circle and even at 3 dimensions, the embeddings were not consistent (see figure 3). So we worked on tweaking the model parameters of CEBRA to see if this would have an impact on the embeddings produced. (See the Appendix section for more details about the Lorenz Attractor.)

2.3 Model Architecture Analysis

The two parameters that seemed to have the biggest impact on the embeddings were the model architecture and the temperature. So we started by looking at the embeddings produced by different model architectures and comparing which ones gave plausible embeddings and which ones did not. This analysis is documented in the Model-arch notebook. Upon subjective viewing of the embeddings produced by the different model architectures, some stood out for different reasons:

[3] The actual dimensionality of the lorenz attractor is ≈ 2.06

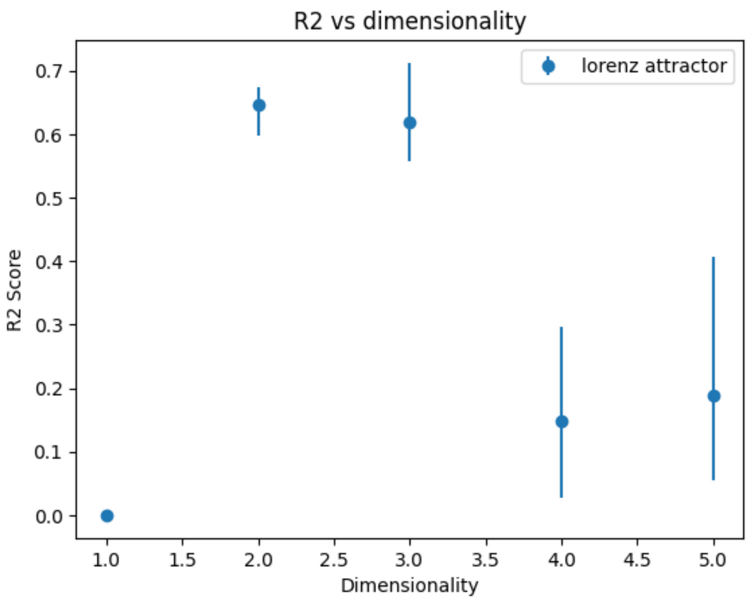

Figure 2: Figure showing the R^2 of the embeddings produced at different output dimensions for the lorenz attractor.

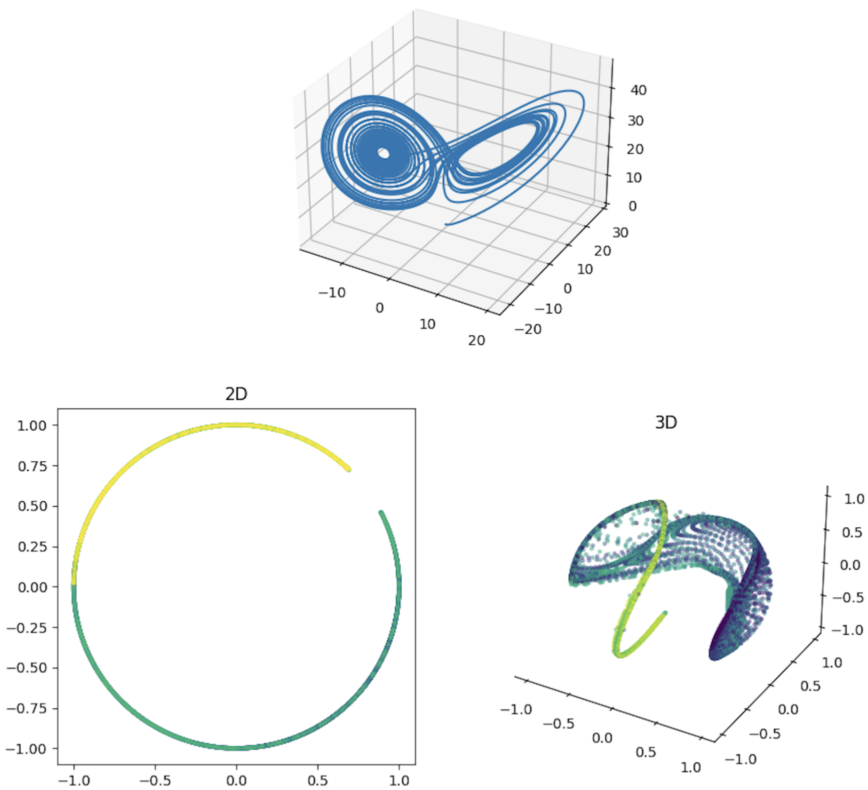

Figure 3: Figure showing the lorenz attractor (top), and its embeddings in 2D and 3D from CEBRA (left and right respectively).

1. offset1-model, offset1-model-mse, offset5-model, offset10-model-mse, offset36-model: for recovering the circles (comparatively) consistently

2. offset10-model, offset-model: for recovering the circles consistently and getting almost plausible representations of the lorenz attractor.

There wasn't a clear standout model, perhaps because we still haven't converged on a set of model parameters that produce the best embeddings, but there's some models that gave very strange embeddings for the lorenz attractor that will be excluded from future analyses.

2.4 Temperature Analysis

The other parameter that visibly impacted the embeddings was the model temperature. In a machine learning model, the temperature is a parameter that gives us a measure of how risk-averse the model will be in producing predictions. For our CEBRA models high temperature values tended to give more unpredictable (but diverse) embeddings. The official documentation for CEBRA defines their temperature parameter as the 'factor by which to scale the similarity of the positive and negative pairs in the InfoNCE loss.' and states that ' higher values yield "sharper", more concentrated embeddings.' [1] Using higher temperatures values of $\approx **$ we were able to occasionally produce plausible embeddings for the lorenz attractor, but replicating them wasn't possible (except for lucky instances).

Another aspect we analysed was using a fixed temperature value versus using a learnable temperature value. CEBRA offers a choice on whether to train the model on different temperature values during the training stage or use a fixed temperature (see figure 4 and figure 5). The learnable temperature gives slightly improved and consistent (i.e. between different initializations) embeddings but the difference is very small.

2.5 Comparison with PCA and MIND

After getting these confusing embeddings from CEBRA we wanted to benchmark them with other established dimensionality reduction methods. If we are able to reproduce these inputs with those other algorithms, then the assumption is that CEBRA should be able to as well. So we began with the PCA, and found that the 2D/3D representations were much closer to the initial inputs than CEBRA was (see figure 6). This means that either CEBRA is doing a bad job at recovering the lorenz attractor or we just haven't found the right parameters yet.

This section will be updated with a comparison with embeddings produced from MIND (Manifold Inference from Neural Dynamics), the manifold embedding algorithm of the Tank Lab.

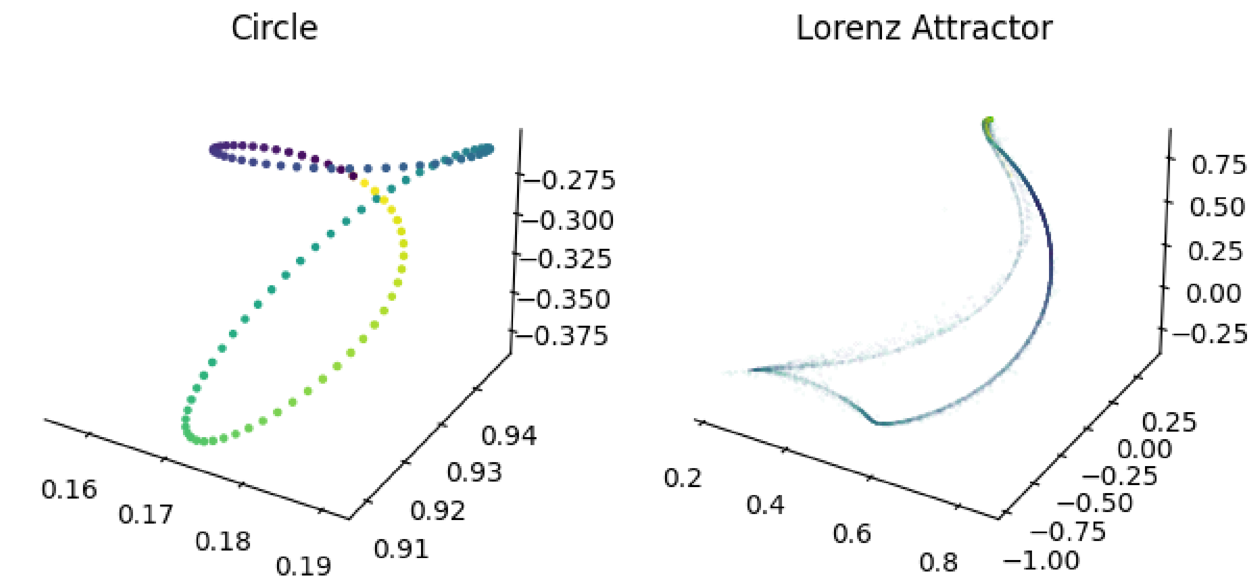

Figure 4: Figure showing embeddings of a circle and lorenz attractor from a model with a fixed temperature parameter

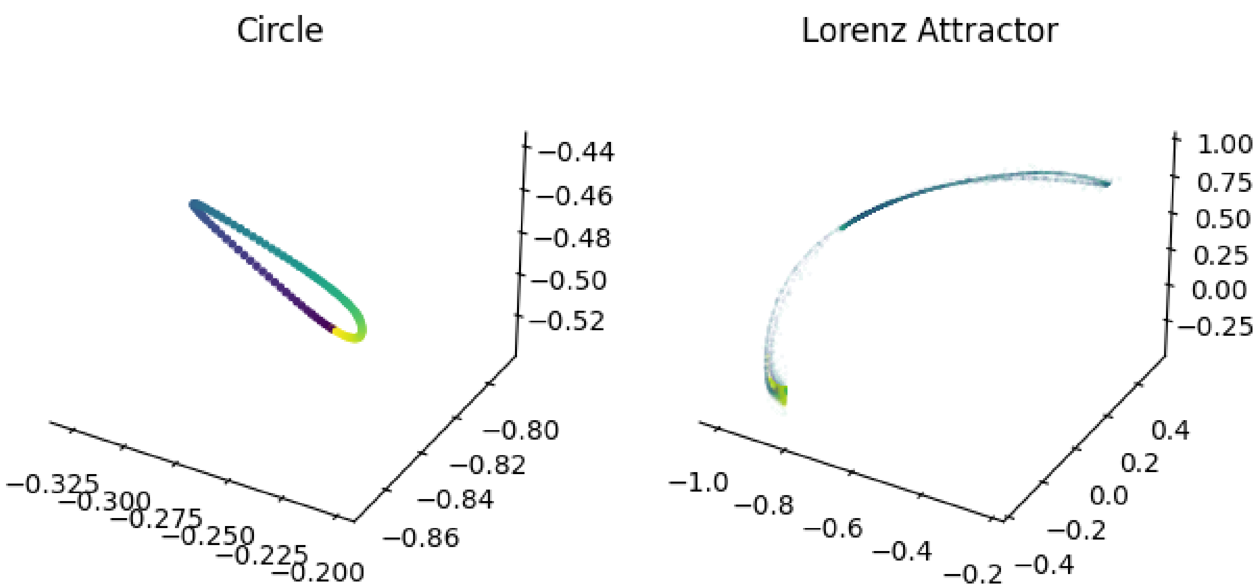

Figure 5: Figure showing embeddings of a circle and lorenz attractor from a model with a learnable temperature parameter

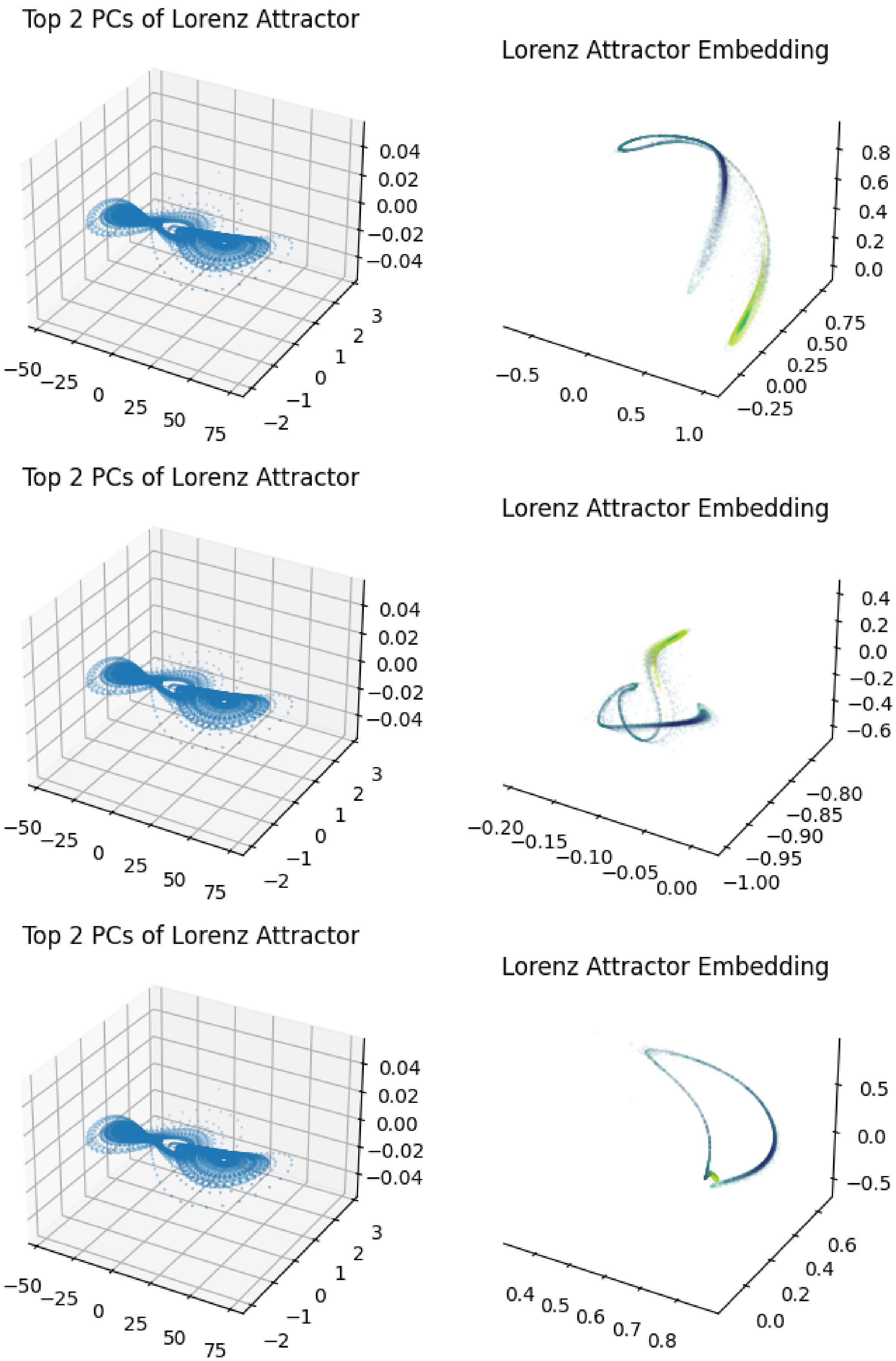

Figure 6: This figure shows a comparison between embeddings of the lorenz attractor produced by CEBRA and the top 2 Principal Components after running PCA on the lorenz attractor. Each row shows a different initialization of the PCA and CEBRA models.

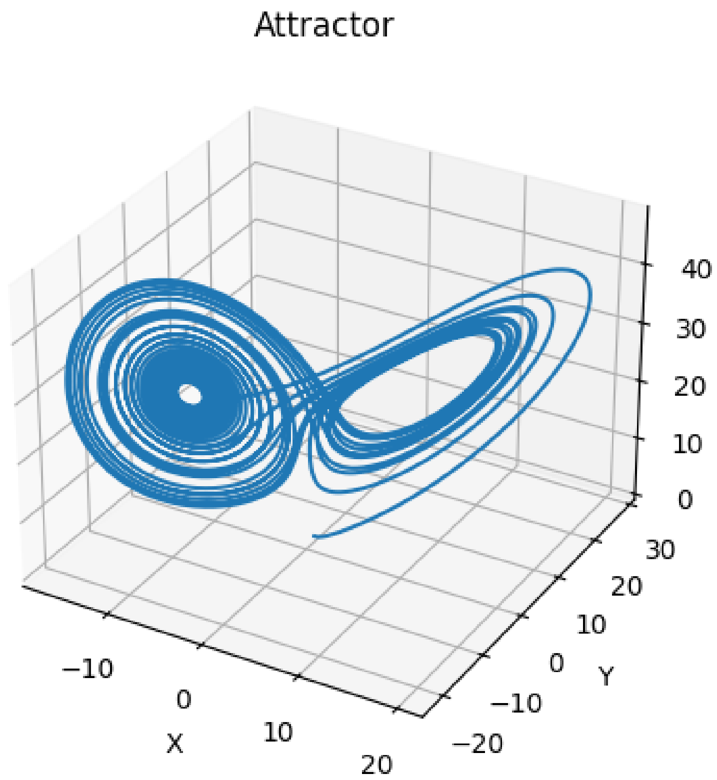

Figure 7: Figure showing the Lorenz attractor - the set of solutions for the lorenz system given some initial conditions and parameter values σ, ρ and β.

3 Appendix

3.1 Lorenz Attractor

The lorenz attractor is a set of chaotic trajectories that are the solution to a system of 3 ordinary differential equations initially used to model climate (see figure 7).

$$\frac{dx}{dt} = \sigma(y - x), \tag{1}$$

$$\frac{dy}{dt} = x(\rho - z) - y, \tag{2}$$

$$\frac{dz}{dt} = xy - \beta z. \tag{3}$$

References

[1] Steffen Schneider, Jin Hwa Lee, and Mackenzie Weygandt Mathis. Learnable latent embeddings for joint behavioural and neural analysis. *Nature*, 617(7960):360–368, may 2023.