Machine Learning Applications in Chaotic Rössler Oscillators

M.Sc. Thesis

By

Nikhil Easaw

Department of Physics
Indian Institute of Technology Indore
June 2022

Machine Learning Applications in Chaotic Rössler Oscillators

A Thesis

Submitted in partial fulfillment of the requirements for the award of the degree

of

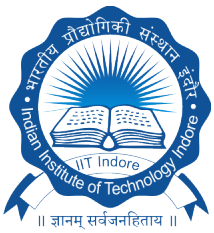
Master of Science

By

Nikhil Easaw

Department of Physics
Indian Institute of Technology Indore

June 2022



INDIAN INSTITUTE OF TECHNOLOGY INDORE

CANDIDATE'S DECLARATION

I hereby certify that the work which is presented in this thesis entitled **Machine Learning Applications in Chaotic Rössler Oscillators** in the partial fulfillment of the requirements for the award of the degree of **Master of Science** and submitted in the **Department of Physics, Indian Institute of Technology Indore**, is an authentic record of my own work carried out during the time period from July 2021 to June 2022 under the supervision of Dr. Sarika Jalan, Professor, Department of Physics, Indian Institute of Technology Indore.

The matter presented in this thesis has not been submitted by me or by anyone else for the award of any other degree of this or any institute.

31/05/22.

Signature of student with date

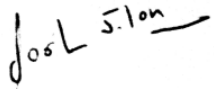
(Nikhil Easaw)

This is to certify that the above statement made by the candidate is correct to the best of my knowledge and belief.

Signature of the Supervisor

Dr. Sarika Jalan

Nikhil Easaw has successfully given his M.Sc. Oral Examination held on **06 June 2022**.



Signature of supervisor of MSc thesis

Convener, DPGC

Date: 27/05/2022

Date: 31/05/2022

Signature of PSPC member

Dr Dipankar Das

Signature of PSPC member

Dr Manavendra Mahato

Date: 30/05/2022

Date: 31/05/2022

ACKNOWLEDGEMENT

Words cannot express my gratitude towards my supervisor Dr Sarika Jalan for allowing me to work on this project. I want to thank her for the invaluable feedback and patience she has shown me. I thank my PSPC members, Dr Dipankar Das and Dr Manavendra N Mahato, for their support and input.

I am grateful to my seniors in CSL for all the discussions and suggestions throughout this journey. I especially thank Prashant Singh Lohiya for his tireless contribution to this project. I thank my classmates for their moral support. Lastly I would like to thank Ayushi Srivastava and Satyabrata Sahoo for the constant motivation to finish this project.

I thank all the nameless staff for providing me with all the wonderful facilities we take for granted.

ABSTRACT

In real-world dynamical systems, partial information about the state of a system is often only available, and predicting the future time evolution of the system from partial time series leads to significant errors in the long term. A correlation matrix of time series is a valuable measure in understanding a system's state. Learning the system's state has many applications in brain research and Earthquake prediction to set up warning systems. Machine learning techniques are popular statistical tools used for prediction in all analytical fields. Using these techniques to predict a system's state from limited time-series information of a few nodes is a novel way to study the system. To this extent, we have devised a model to predict the correlation matrices from limited time series information of a few nodes.

Contents

List of Figures **viii**

1 Introduction **1**

1.1 Representation of a complex system 4

2 Methods and Techniques **7**

2.1 MODEL 7

2.2 Synthetic Data Generation 9

2.2.1 Networks 10

2.2.2 Rössler oscillator 11

2.2.3 Global Synchronization Error 15

2.3 Machine Learning 16

2.3.1 Feed-forward neural network 17

3 Results and Discussions **20**

3.1 Results For the correlation matrix predicted using synthetic data generated 20

3.1.1 Prediction of correlation matrix for varying coupling strengths of same network structure. 21

3.1.2 Prediction of correlation matrix for varying network structure 28

3.2 CONCLUSION 33

3.2.1 UMAP-An Unsupervised Machine Learning algorithm to understand the data set 33

Bibliography **35**

List of Figures

1.1 (a.)Ants forming a bridge without external stimuli (b.) Neurons electrical signals firing in response to one another forming neuronal network 2

2.1 Schematic diagram of model used to predict correlation matrix from partial time series information 8

2.2 3-Dimensional Phase space diagram of a chaotic Rossler oscillator 12

2.3 Time evolution of the x component of Rossler oscillator . . . 13

2.4 (a.)GSE vs λ 100 node ER network (b.)GSE vs λ 100 node SF network 14

3.1 The True and predicted correlation matrix for $N = 50$, $\langle k \rangle = 10$ network displayed for (a) ER network with coupling strength (λ) 0.014. The True and predicted correlation matrix for $N = 100$, $\langle k \rangle = 10$ network displayed for (b) ER network with coupling strength(λ) 0.009 (c) SF network with coupling strength(λ) 0.011 22

3.2 Accuracy of prediction represented by Mean Square Error(MSE) for the correlation matrix predicted, both are done for 100 Nodes with ¡k¿=10 ,(a.)ER random network MSE of predicted correlation matrix versus the length of the time series and number of top degree nodes are increased (b.) ER network MSE of predicted correlation matrix plot for the length of the time series versus number of bottom degree nodes are increased(c.) SF network MSE of predicted correlation matrix plotted for time series length versus the number of top degree nodes used for training (d.) SF network MSE of predicted correlation matrix plotted for length of time series versus the number of bottom degree nodes used for training 23

3.3 Prediction of correlation matrix from 3 different time windows(a,b,c) for 100 node SF network with coupling strength $\lambda = 0.010$ by varying network structure 29

3.4 (a.) Dimension reduction of ER network time series labelled by network structure (b.) Dimension reduction of SF network time series labelled by network structure (c.) Dimension reduction of ER network time series labelled by coupling strength(λ) (d.) Dimension reduction of SF network time series labelled by coupling strength(λ) 32

Chapter 1

Introduction

Complex systems science studies how a large collection of components which are locally interacting with each other at small scales can spontaneously self organize to exhibit global structures and behaviours at larger scale which are not trivial, and often without an intervention of external factors. The properties of the new system may not be understood or predicted from the full knowledge of its constituents alone, such a system is called a complex system, examples include ant hills billions of interacting neurons in the human brain; computers communicating in the Internet; humans in multifaceted relationships. The interaction between the individual components present in a complex system may be of many types with sometimes just a few components involved in the interactions. These interactions make the study of individual components in isolation difficult which leads to difficulty in predicting their future. The main challenge in the field of complexity science is not just to study the individual components but also to understand how these connec-

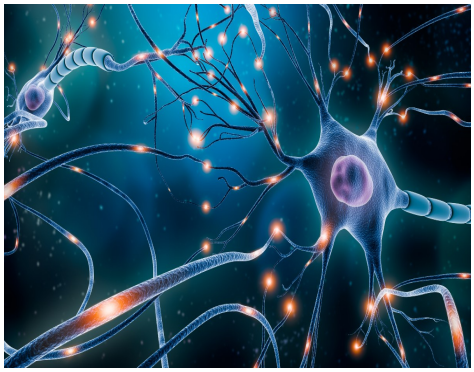


Figure 1.1: (a.)Ants forming a bridge without external stimuli (b.) Neurons electrical signals firing in response to one another forming neuronal network

tions give rise to the whole. The phrase "the whole is more than the sum of its parts" is an ideal way to understand complex systems since the interactions between components causes the system to generate novel information and exhibits non-trivial collective structures and behaviors at larger scales.

Predicting the time series of a dynamical system has many limitations over a long period of time. Since every time step in the time series is a function of the previous time step the error in the prediction of the time series compounds over large time span [4]. To avoid error in prediction we can prefer the correlation matrix of the time series in place of directly trying to predict the time series. A correlation matrix of the time series has several useful applications in real world scenarios. These are discussed below.

There are various applications of constructing the correlation matrix of the time series of nodes, Link prediction methods adopted by various researchers use the correlation matrix of the time series of the nodes, e.g in the field of brain research, MRI or MEG signals are taken from several regions of

the brain and are used to form the correlation matrix which is then used to extract the adjacency matrix(Network structure) by setting a threshold value for the correlation [2]. Further more this simple method of link prediction using correlation strength of time series data has been used to predict the link between two nodes based on the strength of their correlation and then applied to in the prediction of Earthquakes [21]. The correlation between fMRI is used to extract functional states connecting correlated human brain sites [3, 20].

The existing methods only calculate the average correlation matrix of the time series over a fixed period of time. In reality, the correlation matrices of time series vary depending on the length of time as well as the temporal position of the observations. The two main methods for estimating the true covariance matrices are the MLE, maximum likelihood estimation and the GLASSO, graphical least absolute shrinkage and selection operator method. The MLE method assumes a sample covariance matrix from a Gaussian distribution that is then iteratively corrected to estimate the actual covariance matrix by maximizing the likelihood of observing the given time series data. The GLASSO method [7] is an extension of the MLE, except when MLE cannot be applied, for example in the cases where the dimensionality of the Gaussian distribution is higher than the number of observation samples available. This method involves maximizing a log-likelihood function with a penalty term. In[10], the authors have extended GLASSO method to a time-varying GLASSO method where they assume the covariance matrix to be a function of time. Thus, the network structure changes over time with slightly different edges in each time step. In [19], an effective connectivity

network (EC) of the brain regions is formed using the EEG signals from these regions and forming a partial correlation matrix using the GLASSO method. Improving on the existing method, Ref [12] introduces two new estimations methods called P-GLASSO and DP-GLASSO.The prerequisite of all the above techniques involves having knowledge of the time series of all the nodes of the network, whereas in real-world networks, this information is rarely available.

Traditional approaches like stacking of algorithms [6], collecting data from a set of core nodes and then using traditional metrics such as common neighbour score and optimizing the number of fringe nodes data required for the best prediction[1] or developing other similar metrics[14] and using its score as an indicator of links. Determining the most connected nodes is also a difficult task in biological networks. Thus, these methods fail to model such cases where the number of nodes is much higher than the number of time series available. To this point, we hereby explore a machine learning-based approach using the time series of a few nodes to reconstruct the correlation matrix of the entire network. Thus, our method relies only on the time series of a few nodes, which is available in the real-world networks.

1.1 Representation of a complex system

A dynamical system with N nodes can be represented in terms of a time series matrix **T** in which the time evolution of each node in the system is recorded in the rows of the matrix and each column representing the indi-

vidual nodes/components of the system. In matrix terms

$$\mathbf{T} = \begin{bmatrix} t_{11} & t_{12} \dots & t_{1N} \\ t_{21} & t_{22} \dots & t_{2N} \\ \vdots & \vdots \ddots & \vdots \\ t_{m1} & t_{m2} \dots & t_{mN} \end{bmatrix}$$

where T_{ij} represents the time-series information of the j^{th} node at the i^{th} time step. The amount of influence of one node on another in a complex system can be represented in terms of the correlation between their time series. A linear measure, Pearson correlation coefficient is used since it is the most relevant one in our case. The Pearson correlation coefficient between the time series of nodes is then given by an $(N * N)$ matrix $\mathbf{R}$, where R_{ij} represents the correlation between the time series of i^{th} node and j^{th} node. We predict the correlation matrix $\mathbf{R}$ of the entire time series $\mathbf{T}$ from a subset $\delta\mathbf{T}$ of the original time series. The whole correlation matrix

$$\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} \dots & r_{1N} \\ r_{21} & r_{22} \dots & r_{2N} \\ \vdots & \vdots \ddots & \vdots \\ r_{N1} & r_{N2} \ddots & r_{NN} \end{bmatrix}$$

The correlation measurement used is the linear the Pearson correlation coefficient r [6] given as

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2(y_i - \bar{y})^2}} \tag{1.1}$$

with x_i and y_i representing the time evolution data of each node, and t_{ij} is the time evolution of a node at each time step. The number of input time steps k is always less than the total time m step used for creating the correlation matrix and a subset of the time series $\delta\mathbf{T}$ predicts the correlation matrix for the entire system.

Chapter 2

Methods and Techniques

The techniques used for preparing the synthetic data and the machine learning algorithm is discussed in details in this chapter.

2.1 MODEL

The model predicts the cross-correlation matrix of the entire time series from limited time-series information of a few nodes. For this purpose, we use a machine learning algorithm feed-forward neural network to predict the correlation coefficients. The input given to the neural network is a subset of the time-series matrix of the whole system, and the output is the upper triangular portion of the correlation matrix. The subset predicting the correlation matrix is prepared in the shape of a column vector with information of each

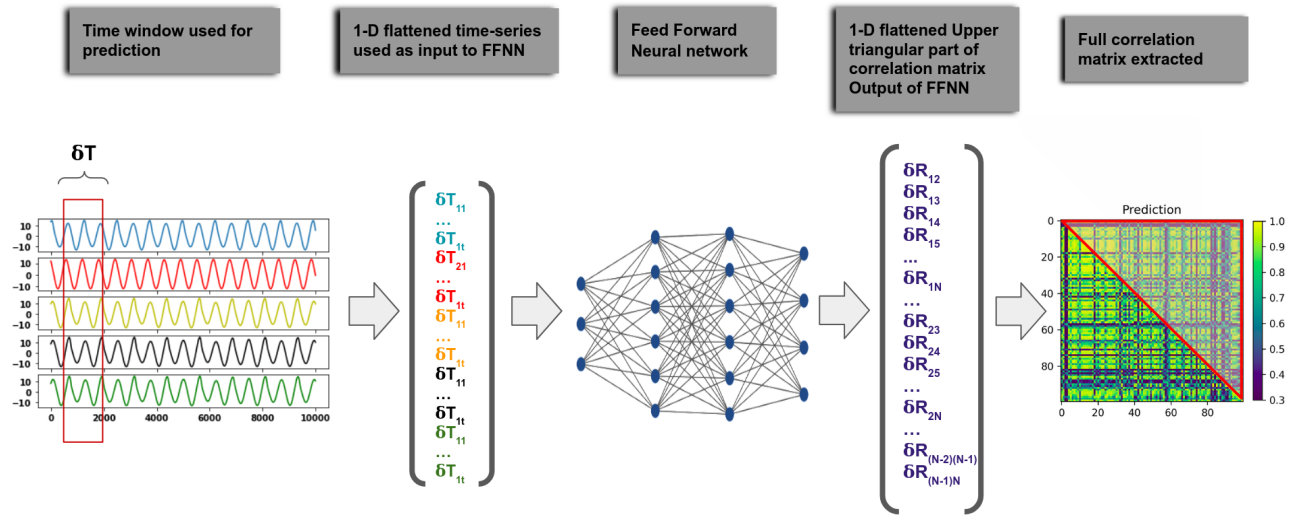


Figure 2.1: Schematic diagram of model used to predict correlation matrix from partial time series information

node used stacked one on top of each other to form the vector. The output from the neural network is also a column vector of the upper triangular portion of the correlation matrix. The output vector contains correlation coefficients on top of each other row-wise. The entire correlation matrix is computed using the symmetry of the matrix with diagonal elements equated to 1 since they are self correlations.

The neural network used in the model is trained using time series and correlation matrices similar to the ones used for prediction. The data set used for training the neural network contains the time series and correlation matrices generated from networks having the same number of nodes and average degree as the test data. The split between training and testing data is explained in detail in the result sections. The time series generating a correlation matrix is divided into several time windows depending on the

length of the time series intended to predict the correlation matrix. All the time windows generated from a time series are used for training its respective correlation matrix. The correlation matrix $\mathbf{R}$ is predicted from the matrix $\delta\mathbf{T}$ with elements δt_{ij} but here $i \in (1 \dots k)$ where $k < m$ and $j \in (1 \dots n)$ where $n < N$.

$$\delta\mathbf{T} = \begin{bmatrix} \delta t_{11} & \delta t_{12} \dots & \delta t_{1n} \\ \delta t_{21} & \delta t_{22} \dots & \delta t_{2n} \\ \vdots & \vdots \ddots & \vdots \\ \delta t_{k1} & \delta t_{k2} \dots & \delta t_{kn} \end{bmatrix}$$

The correlation matrix $\widetilde{\boldsymbol{R}}$ predicted from the machine learning algorithm is compared to the true correlation matrix $\mathbf{R}$ and Mean Square Error(MSE) is used as the measure of accuracy of the predicted correlation matrix($\widetilde{R} \approx \mathbf{R}$).

2.2 Synthetic Data Generation

We use two random graph models, the Erdos-Renyi (ER) and Scale-Free (SF) networks, to model the dynamical system. The coupled dynamics of the nodes on these graph models generate the system's times series. The ER random network with N nodes and average degree $\langle k \rangle$ is generated by: starting with N number of nodes, connecting all pairs of nodes with a probability $p = N/\langle k \rangle$. The ER random network thus generated will have a

Gaussian degree distribution. The SF networks generated follows a power-law degree distribution using the Barabasi-Albert model. The nodes model chaotic Rössler oscillator with nodes that are parameter mismatched.

2.2.1 Networks

A network is a static representation of a complex system. The individual components in a complex system are called nodes or vertices and if there is a direct interaction between the components they are represented by links or edges. Examples of networks include social networks, biological networks, technological networks etc. The structural property used to create networks for our purpose is the average degree which is defined as

Average degree : Degree of a node is the number of edge(E) connected to a node. The average degree of a network is thus

$\langle k \rangle = \frac{2E}{N}$

where N is the total number of nodes and E total number of edges.Based on degree distribution networks can be classified into different types, the two most popular models are Erdos-Renyi random(ER) and Scale Free(SF) networks.

ER random networks:

ER random network coined by Erdos Renyi. We generate a random network from ER model given two parameters for construction. The number

of nodes(N) present in a network and probability p that a given node is connected to another node.

The average degree of the network will be $\langle k \rangle$=p*(N-1)

Scalefree networks

Albert Barabasi proposed the scalefree model, based on preferential attachment algorithm. They are constructed using BA preferential method where each node prefers to connect with higher degree nodes. The nodes with higher degree has more chance to connect with incoming nodes. Most real world networks are scale free in nature.

Scale free networks follow power-law degree distribution.

2.2.2 Rössler oscillator

The time evolution of the phase of each node in the network is modelled by the **x** coordinate of the Rössler oscillator[18] in phase space whose equation is given by:

$$\begin{aligned} \dot{x_i} &= -\omega_i y_i - z_i + \lambda \sum_{j=1}^{N} A_{ij}(x_j - x_i) \\ \dot{y_i} &= \omega_i x_i + 0.15 y_i \\ \dot{z_i} &= 0.2 + z_i(x_i - 10) \end{aligned} \tag{2.1}$$

where $i = 1, \ldots, N$ and N is the number of nodes of the network, ω_i is the natural frequency of i^{th} node, the intrinsic frequency distribution for the nodes

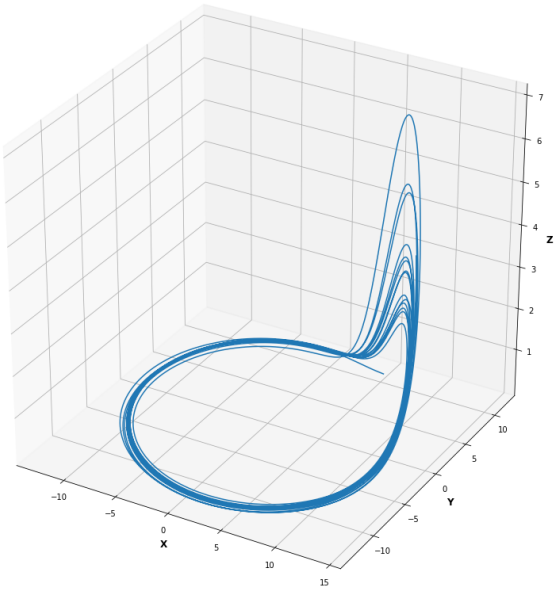


Figure 2.2: 3-Dimensional Phase space diagram of a chaotic Rossler oscillator

in the oscillator is drawn from a normal degree distribution with mean 1 and variance 0.03 this frequency is given by $\omega + \Delta\omega$ here the $\Delta\omega = 0.03 * k$ where k is generated from a normal distribution. The frequency is the mismatched parameter in the set of oscillators and λ denotes the overall coupling strength between the connected nodes. The connectivity structure of the nodes is represented by an adjacency matrix A_{ij}, where $A_{ij} = 0$ represents there is no connection between the nodes and $A_{ij} = 1$ represents that the nodes i and j are connected.

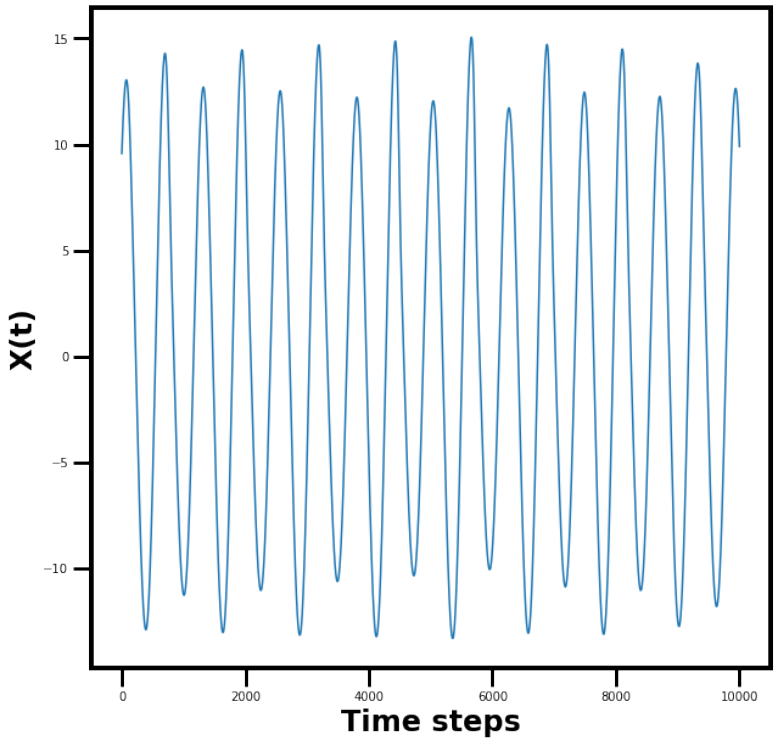


Figure 2.3: Time evolution of the x component of Rossler oscillator

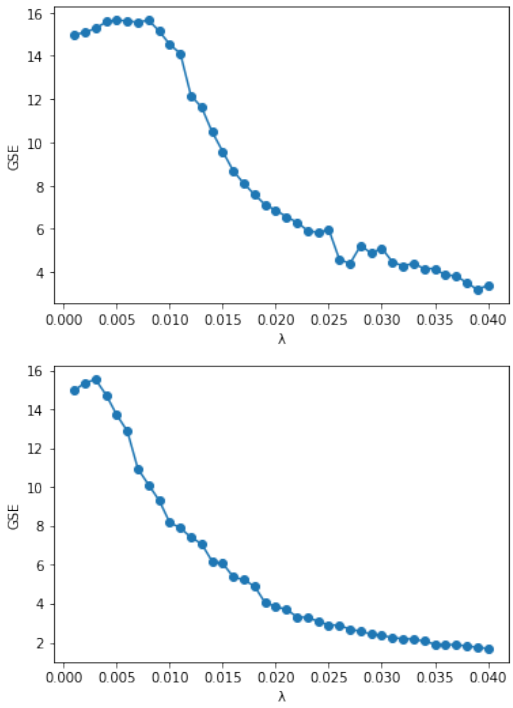


Figure 2.4: (a.)GSE vs λ 100 node ER network (b.)GSE vs λ 100 node SF network

2.2.3 Global Synchronization Error

The synchronization level of the system is measured using Global Synchronization Error (ΔE) of the time series of the nodes over the time window of measurement . The ΔE of the time series matrix $\mathbf{T}$ of order $m \times N$ is calculated using using the following formula

$$\Delta E_i = \frac{1}{m(N-1)} \sum_{j=1}^{N} \sum_{t}^{t+m} ||x_j - x_i|| \tag{2.2}$$

gives the error of the i^{th} node where $||x||$ represents $\sqrt{x^2 + y^2 + z^2}$ where x,y,z are phase space coordinates of the Rössler oscillator. This error is computed for all N nodes and their average value is the Global Synchronization Error of the time series of a particular coupling strength (λ).

The graph of ΔE vs λ 2.4 shows that the system transitions towards synchronization with increase in coupling strength between the nodes of the networks. The semi-synchronous region of the system is determined from the Fig.12.4 coupling strengths corresponding to the middle portion of the ΔE vs λ plot is treated as the semi-synchronous region of the dynamical system, for Rössler oscillator this corresponds to region with ΔE lying between 12-8 . The time series (phase evolution of the nodes with time) is then recorded for a few values of such coupling strengths after an initial transient. We only use one of the three phase space variables to represent the time series of one of the nodes of the dynamical system.

The value of coupling strengths(λ) corresponding to semi-synchronous

region of the network of coupled chaotic oscillators is used to predict the correlation matrix, we choose this region for the prediction of correlation matrix since in the completely asynchronous region with high values of ΔE and low coupling strength(λ) there would be no correlation between the time series of the nodes to predict anything meaning full and in the highly synchronized region the correlations would be 1 and there isn't any need to predict the correlations between the time series of the nodes. In the semi-synchronous region after a limiting coupling strength(λ) coupled chaotic oscillators with mismatched parameters reach the generalized synchronization regime this corresponds to the region where the predictions for correlation matrix of the time series show greater accuracy.

We solve the coupled Rössler oscillator equations using a 4^{th} order Runge-Kutta method with a step size of $h = 0.01$. The phase values $x(t)$ in eq. (3.1) of each node were recorded for selected values of coupling strength (λ) for which the nodes were in semi-synchronous region as measured from the values of Global Synchronization Error (ΔE).

2.3 Machine Learning

When a computer learns anything without being explicitly coded it called as machine learning. A range of mathematical algorithms learn from training data to find an optimal relationship between the input variables and output variable, The relationship acts as a function which maps the input to an unknown output. There are many machine learning algorithms which can

be broadly divided into two categories (a.) Supervised machine learning algorithms (b.) Unsupervised learning algorithms.

Supervised machine learning algorithms : These are algorithms in which an output is expected from the input variables. The output expected is first trained to the machine learning algorithm. Examples Support vector machines, Linear Regression, Feed Forward Neural Networks.

Unsupervised machine learning algorithms : These are algorithms which are not explicitly trained to give any output. They classify the input data based on their properties. Examples are UMAPS, K-means clustering, Hierarchial clusterring

In our survey we found that the Feed Forward neural network is most appropriate for our purpose.

2.3.1 Feed-forward neural network

We use a feed-forward neural network (FNN) for predicting correlation matrix from the time series of a few nodes. Our task is a supervised learning with inputs of time series of a few nodes, and outputs of corresponding correlation matrix. Thus, we assign (number of nodes$\times$ length of time series) neurons for the input layer, and $\frac{N^2}{2} - N$ nodes for the output layer. **Note that the input to the neural network is the small subset of the time series data corresponding to the nodes and output layer is the predicted correlation matrix for the entire network.** We adopt the SELU (Scaled Exponential Linear Unit) as a basic activation function except for the

output layer[15]. For the output layer, we use sigmoid function, since correlation value is between 0 and 1. We use Adam (Adaptive Moment Estimation algorithm) for the neuranl netwok optimization [14]. We construct a model through one input layer, two hidden layers, and one output layer. A layer is composed of several neurons, and neurons in adjacent layers are connected to each other. The relationship between the input($a_j^{(l-1)}$) and output($a_i^{(l)}$) of a layer is as follows:

$$z_i^{(\ell)} = \sum_{j=1}^{K} w_{j,i}^{(\ell-1)} a_j^{(\ell-1)}, \tag{2.3}$$

$$a_i^{(\ell)} = \mathrm{SELU}(z_i^{(\ell)}). \tag{2.4}$$

where $\omega_{i,j}$ is a weighted connection between the j^{th} neuron of $(l-1)^{th}$ layer and i^{th} neuron of $(l)^{th}$ layer and K is the number of neurons in the $(l-1)^{th}$ layer. The $a_j^{(l)}$ indicate output of j^{th} neuron in the l^{th} layer. SELU(x) is an activation function, introduced in [15]. The neural network receives an input $(a^{(1)})$ and generates an output $(a^{(L)})$ through the above propagation rule. Training a neural network means finding w that can give us the desired output for the input. This is the same as the process of reducing the difference between the neural network output $a^{(L)}$ and the desired output Y. We define the difference as follows and call it the loss function.

$$Loss = -\sum_{k=1}^{N_L} (Y_k log(a_k^{(L)}) + (1 - Y_k) log(1 - a_k^{(L)})), \tag{2.5}$$

We use Adam algorithm [14] for the minimizing the loss function.

Chapter 3

Results and Discussions

3.1 Results For the correlation matrix predicted using synthetic data generated

The results of correlation matrices predicted from the synthetic data generated using coupled Rossler oscillators are presented below. The success of our model in predicting the correlation matrix from a subset of time series is shown by splitting the data set used for training and testing in two ways. In the first case, we split the data set by the coupling strength, and in the second case, we split the data set using network structure. The training and test data set is explained in detail, along with the results of each case. Why the machine learning algorithm predicts correlations is exaplained using an unsupervised machine learning algorithm called UMAP. Both cases are tested on ER and SF networks.

3.1.1 Prediction of correlation matrix for varying coupling strengths of same network structure.

The time-series data used to train and test the model differs by coupling strength (λ). The subset of time series $\delta\mathbf{T}$ used to test the model is generated from the network structure used to train the model with a different coupling strength.

Data Set for Rössler oscillators : 75 different network structures with the same average degree $\langle k \rangle = 10$ and number of nodes $NN = 100$ is generated for both ER and SF networks. 5 coupling strengths, $\lambda = \{0.012, 0, 013, 0.014, 0.015, 0.016\}$ for ER and $\lambda = \{0.006, 0.007, 0.008, 0.009, 0.010\}$ in the semi synchronous region (Ref appendix) is selected for generating the time series and their corresponding correlation matrix. Time series generated from 4 of the coupling strengths is used to train the model. The time series generated from the 5^{th} coupling strength is used to test the model . The ER and SF networks are trained and tested separately.

Training data set : The training data set used contains $300(75 * 4)$ different time series and its respective correlation matrix to train the model. The 4 coupling strengths selected are varied. So that predictions can be made for each coupling strength.

Test data set : The test data set used contains $75(75*1)$ time series with the same coupling strength corresponding to different network structures.

The predictions for the correlation matrix for $NN = 100$ ER and SF with average degree $\langle k \rangle = 10$, figure 3.1 shows the true correlation matrix

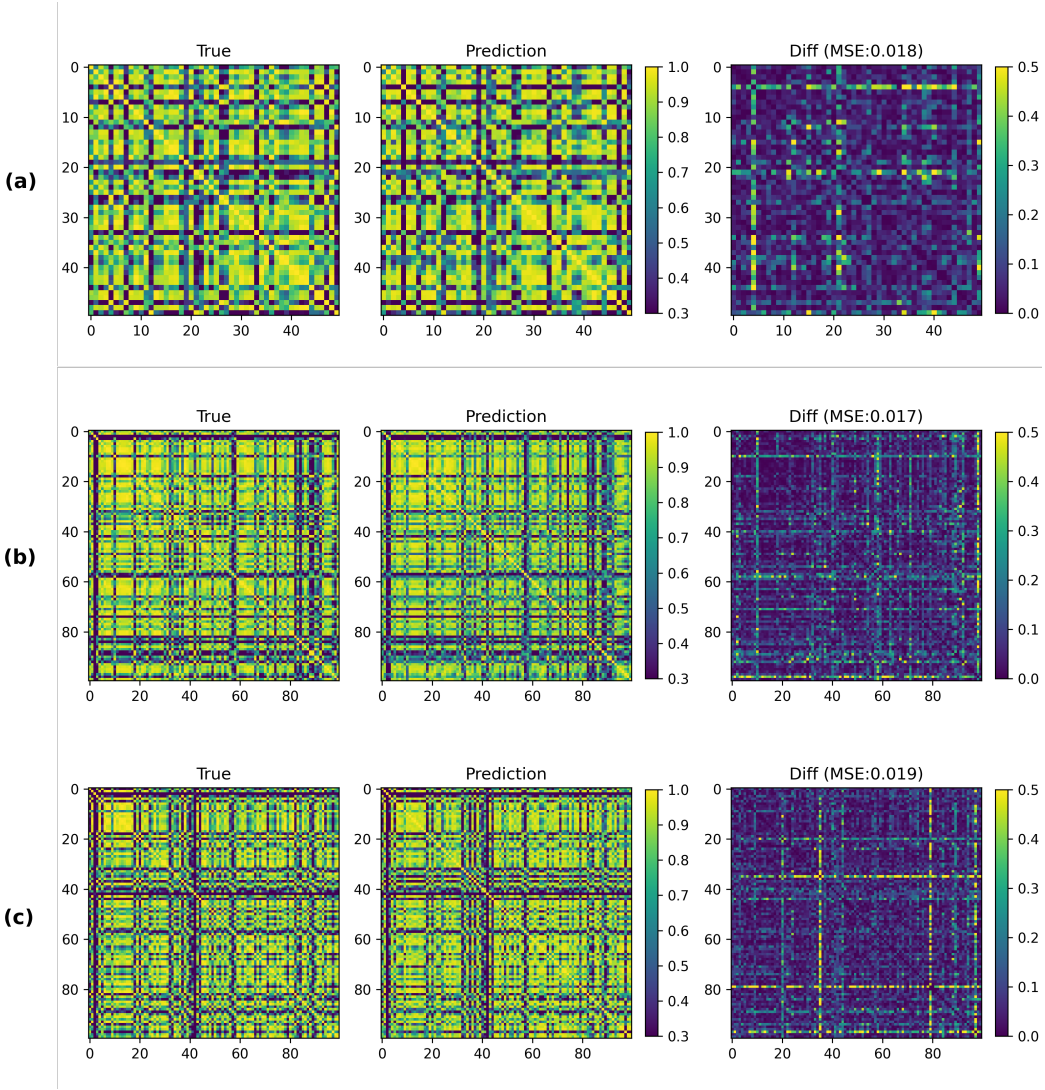


Figure 3.1: The True and predicted correlation matrix for $N = 50$, $\langle k \rangle = 10$ network displayed for (a) ER network with coupling strength (λ) 0.014. The True and predicted correlation matrix for $N = 100$, $\langle k \rangle = 10$ network displayed for (b) ER network with coupling strength(λ) 0.009 (c) SF network with coupling strength(λ) 0.011

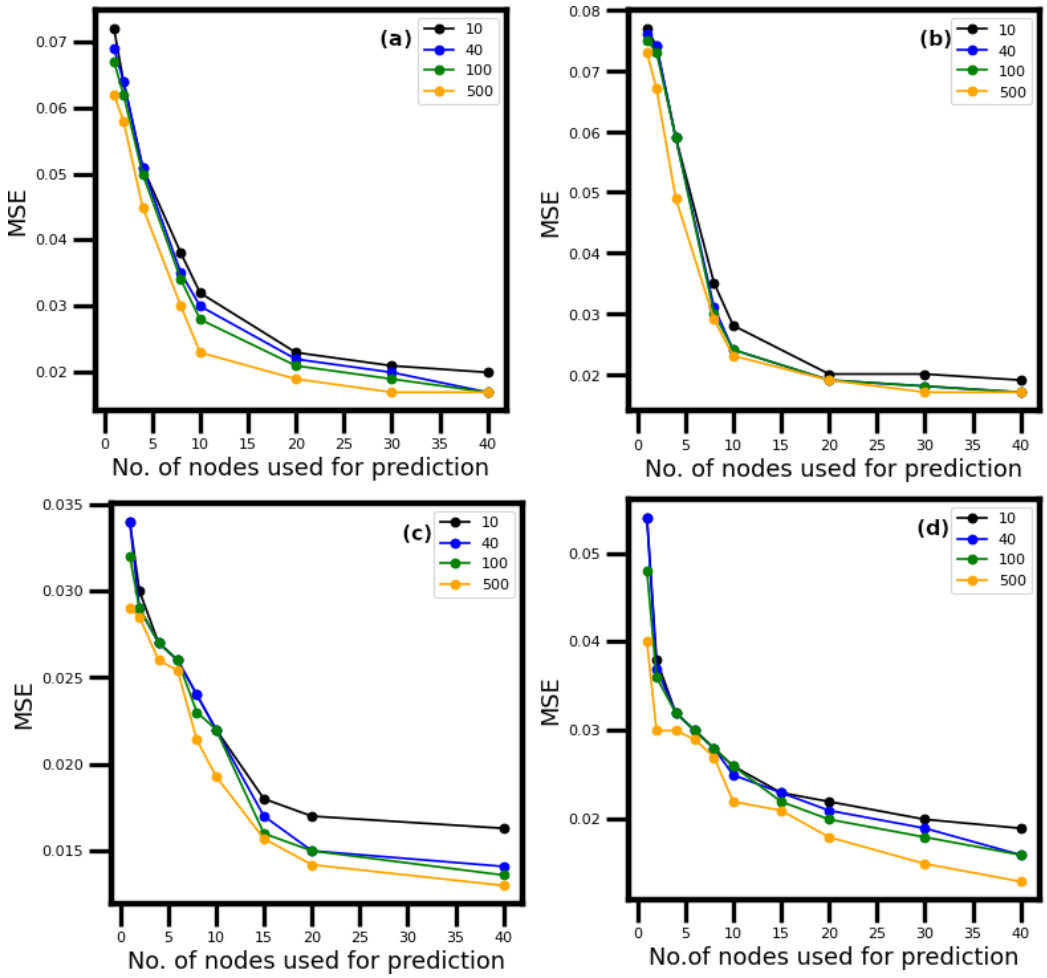
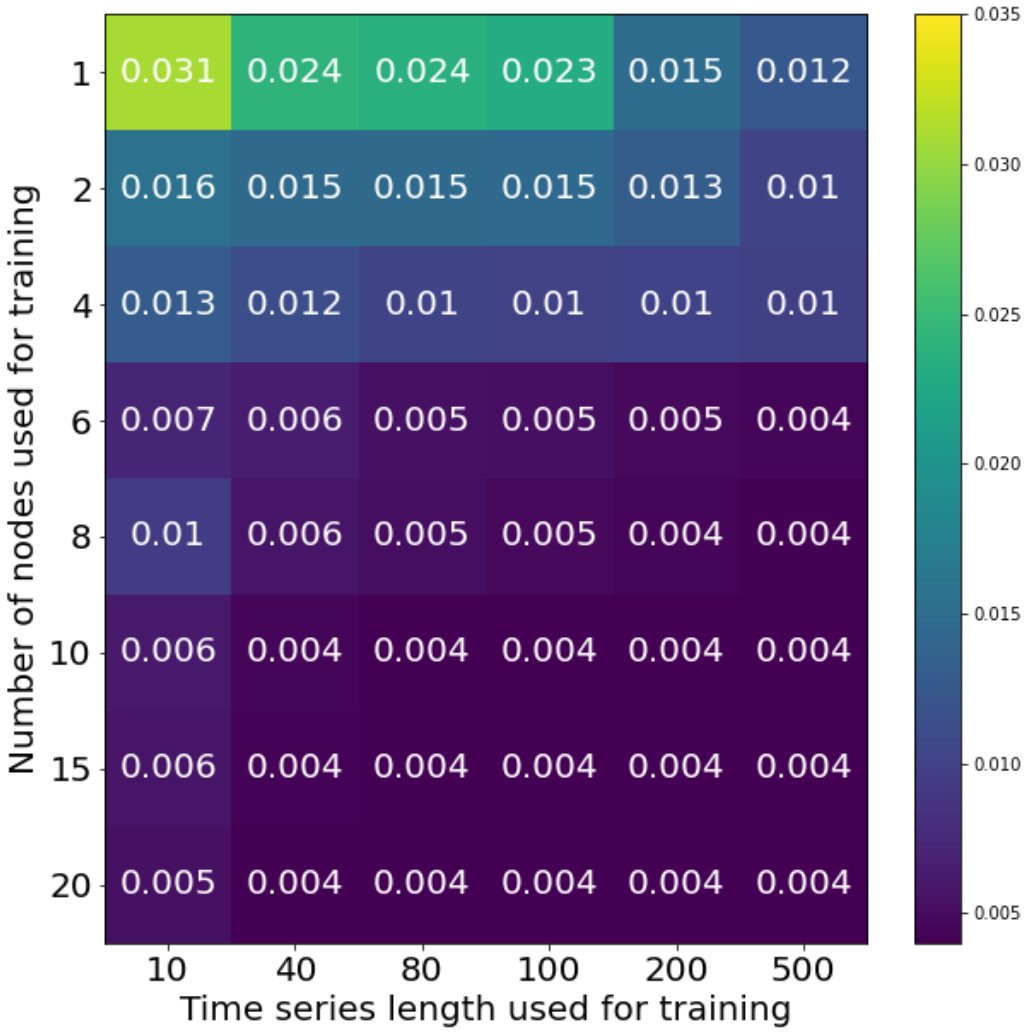


Figure 3.2: Accuracy of prediction represented by Mean Square Error(MSE) for the correlation matrix predicted, both are done for 100 Nodes with ¡k¿=10 ,(a.)ER random network MSE of predicted correlation matrix versus the length of the time series and number of top degree nodes are increased (b.) ER network MSE of predicted correlation matrix plot for the length of the time series versus number of bottom degree nodes are increased(c.) SF network MSE of predicted correlation matrix plotted for time series length versus the number of top degree nodes used for training (d.) SF network MSE of predicted correlation matrix plotted for length of time series versus the number of bottom degree nodes used for training

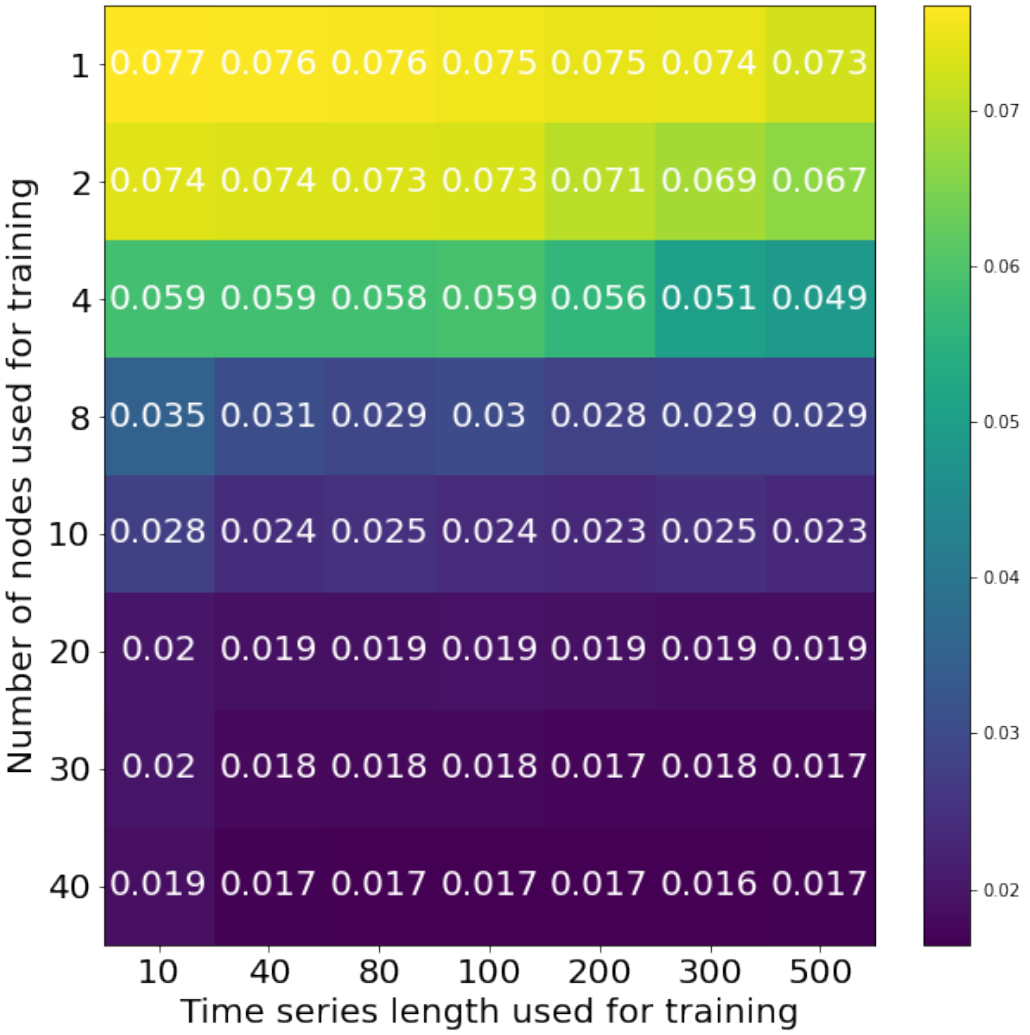
along with the predicted correlation matrix and the difference between the predicted and the true correlation matrix.

The effect of number of nodes(NN) and length of time series used in predicting the correlation matrix is observed by systematically increasing both and repeating the experiment. The accuracy of the prediction is measured by the Mean Square Error(MSE) of the difference between the predicted correlation matrix $\delta\mathbf{R}$ and the true correlation matrix $\mathbf{R}$. The MSE of prediction is calculated by averaging MSE from each coupling strength. A performance array [Refer appendix]is by varying the NN and length of time series used in prediction. The predictions are made for the time series of high degree nodes and low degree nodes of the network. From observing the performance array we see time series generated from ER networks high degree nodes perform slightly better as compared to lower degree nodes. In SF networks the performance of both high degree nodes and low degree nodes are similar. The NN has more influence on the accuracy of prediction as compared to the length of time series. The increase in accuracy of prediction with the increase in NN and length of time series used reaches a saturation this is observed from 3.2. Since we consider higher degree as well as lower degree nodes we assume the random nodes will also follow the same properties of convergence.

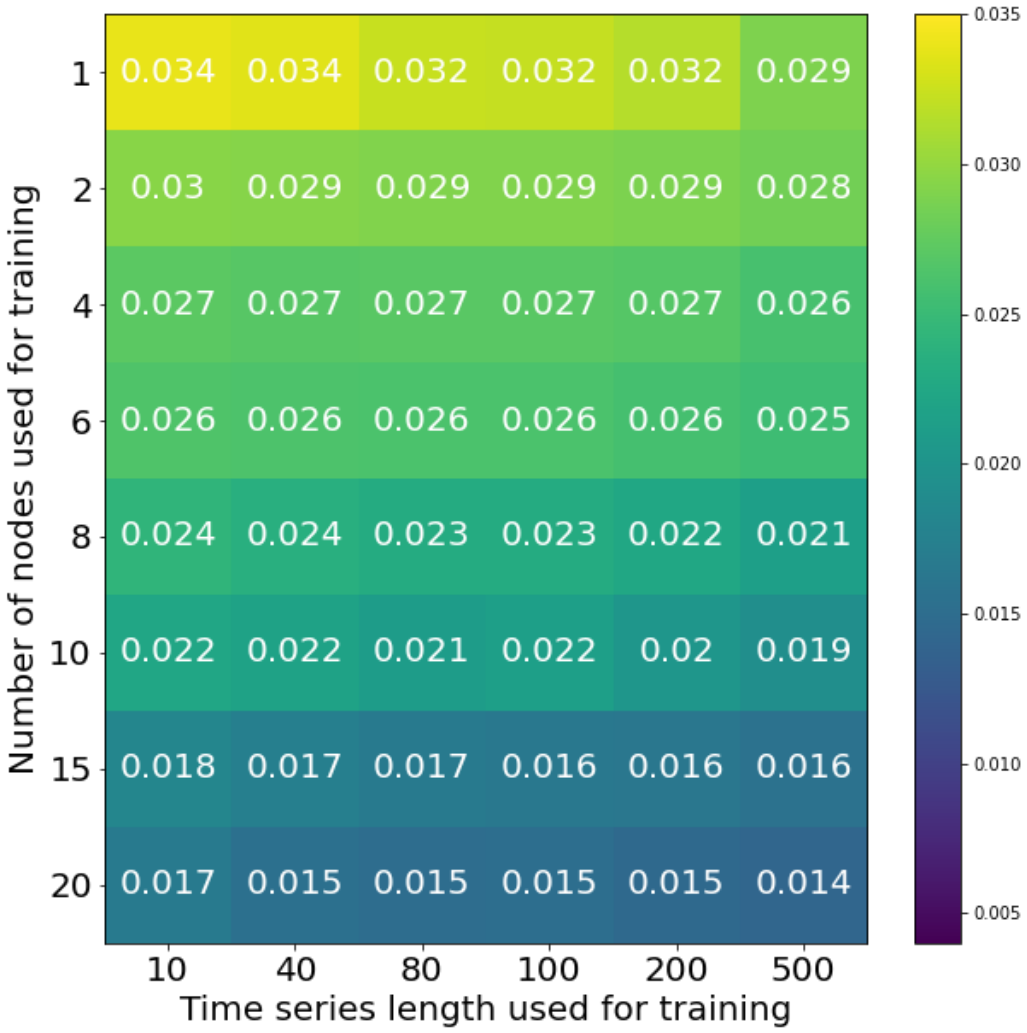
Performance array of the model for varying length of time series and nodes used for prediction



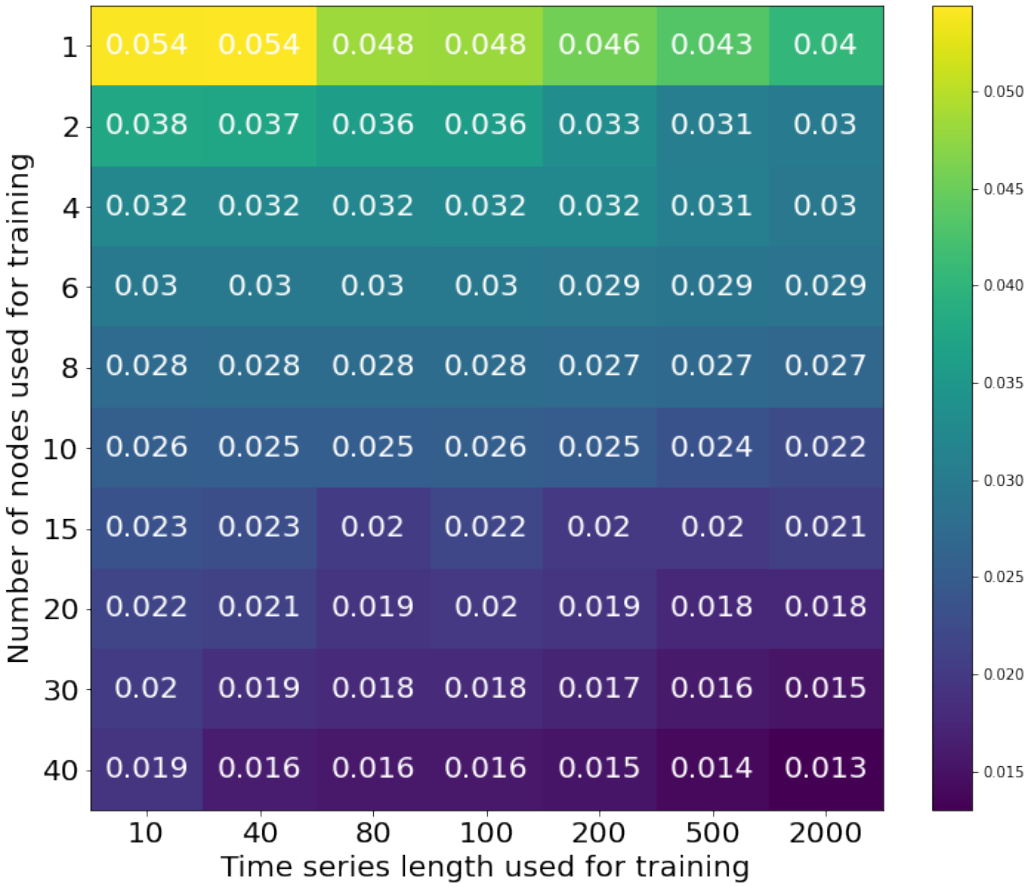
ER random Network with nodes with high degree distribution



ER random Network with nodes with low degree distribution



SF Network with nodes with high degree distribution



SF Network with nodes with low degree distribution

3.1.2 Prediction of correlation matrix for varying network structure

The same model is then applied to predict the correlation matrix when the information of the underlying network structure is not used for training model. To train the model we use time series and correlation matrix data which show close resemblance to the testing data, for this we generate data from

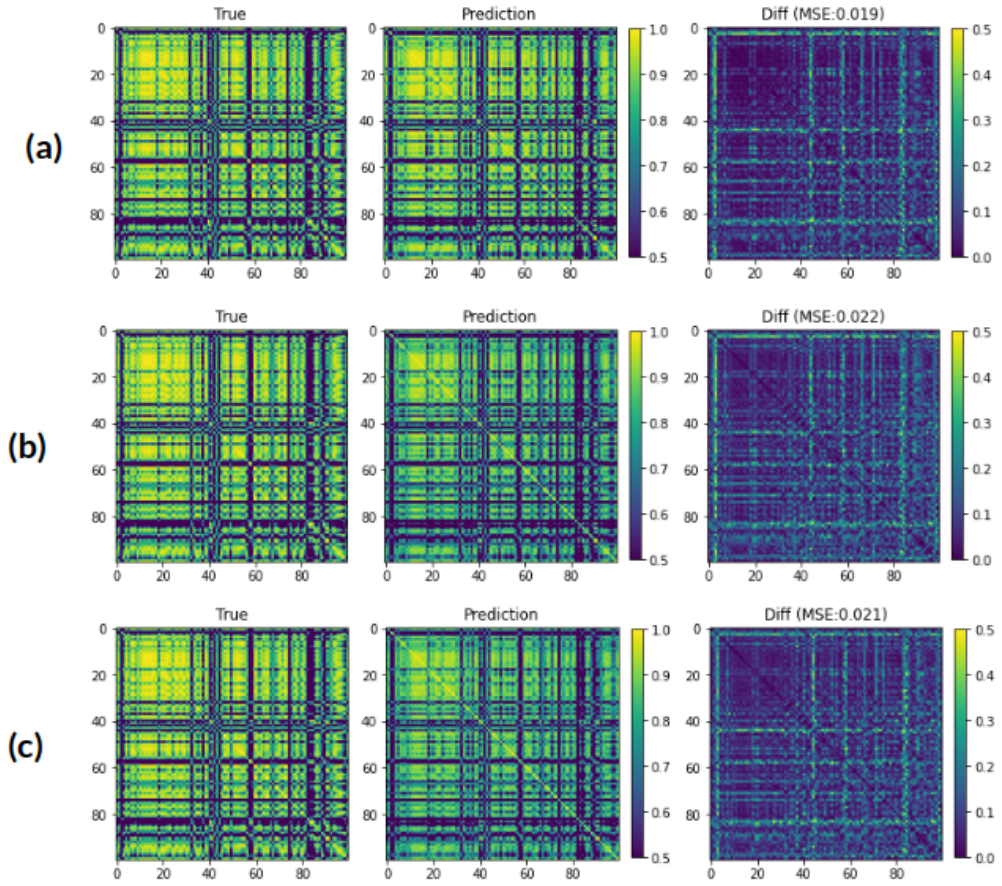


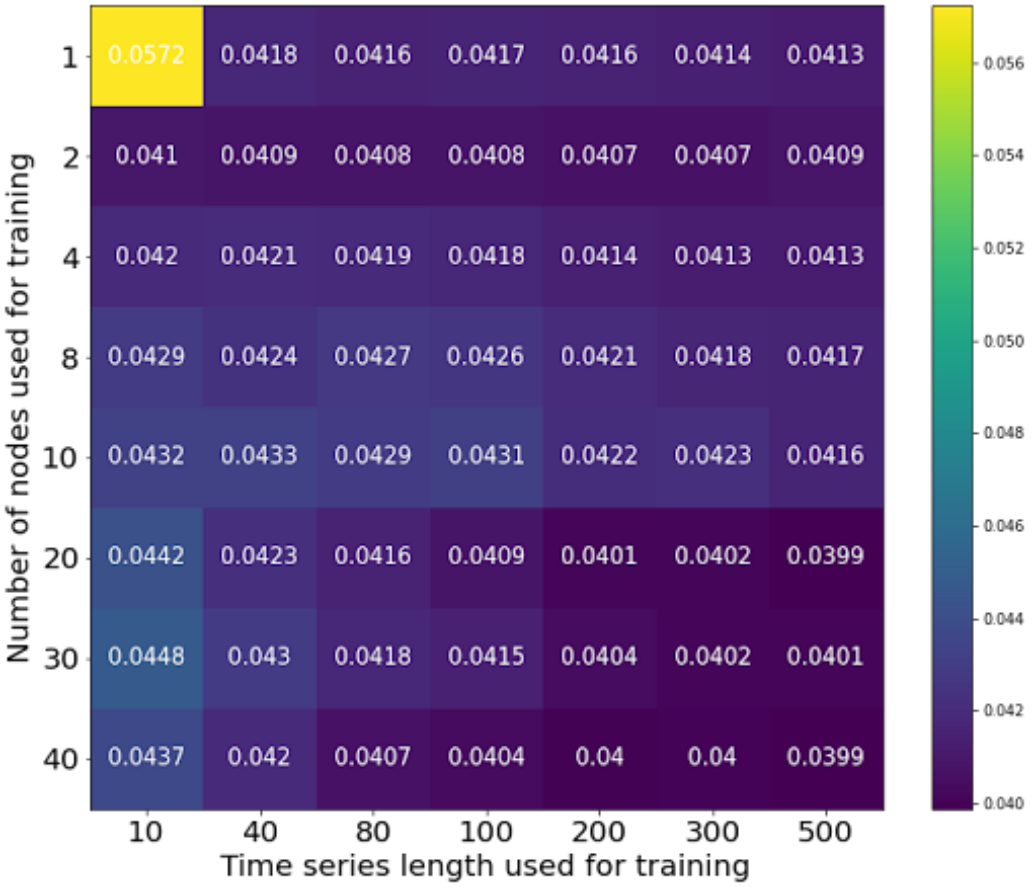
Figure 3.3: Prediction of correlation matrix from 3 different time windows(a,b,c) for 100 node SF network with coupling strength $\lambda = 0.010$ by varying network structure

different network structures with the same average degree as the test network structure. We train the model for high degree nodes and low degree nodes for both ER and SF networks. The number of nodes(NN) and the length of time series used for predicting the correlation matrix is increased systematically to understand their effects on prediction. The MSE vs NN(Number of Nodes used for prediction) is is stored as a performance array. From the prediction results we find that SF networks show good results for the varying network structure. To explain the result we use an unsupervised learning algorithm UMAP to explain the results of both the ER and SF network. The data set used is the same as the previous case with the difference in the split between training and testing data set.

Training Data Set: The Training Data set is the time series generated by 74 different network structures the same average degree $\langle k \rangle$ and Number of nodes for the same coupling strengths as the previous case

Testing Data Set: The test data set is the time series generated by the 75^{th} network structure for the same coupling strengths.

Performance array of the model for varying length of time series and nodes used for prediction



SF Network with nodes with high degree distribution

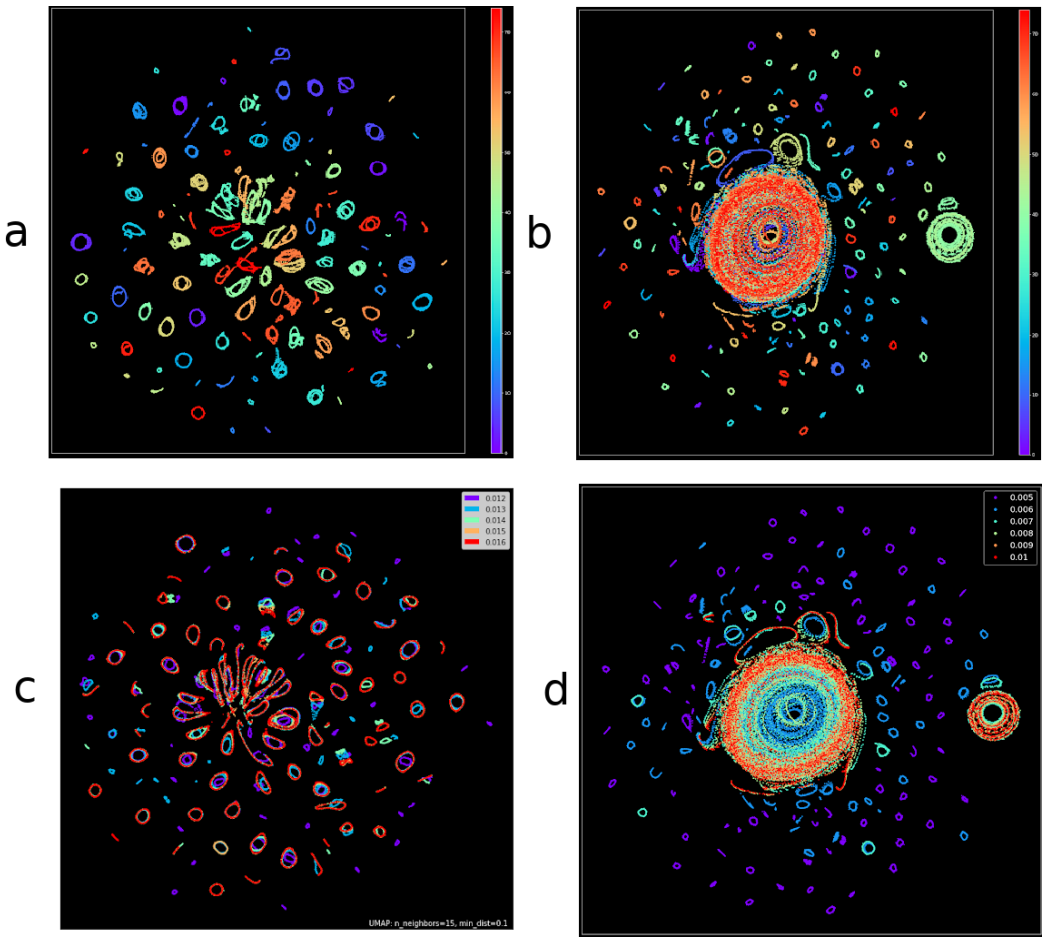


Figure 3.4: (a.) Dimension reduction of ER network time series labelled by network structure (b.) Dimension reduction of SF network time series labelled by network structure (c.) Dimension reduction of ER network time series labelled by coupling strength(λ) (d.) Dimension reduction of SF network time series labelled by coupling strength(λ)

3.2 CONCLUSION

3.2.1 UMAP-An Unsupervised Machine Learning algorithm to understand the data set

We use an unsupervised machine learning approach to understand the intrinsic properties of the data set. The input to the UMAP is time series subsets from one window of each coupling strength used in the model. UMAP transforms the time series column vector used for prediction to points in a 2-dimensional space. The distance between the points in this 2-Dimensional space represents the difference between the various subsets used for prediction. The points forming clusters together are similar in property; in such cases, a set of points which are part of the cluster, even if not included in the training dataset, can be extrapolated from the cluster.

Time series generated by coupled Rössler Oscillator : The time series subset from ER random network created using coupled Rössler oscillators clusters together by network structure, so the model predicts a missing coupling strength from the same network structure from this data set. The clusters formed by the different network structures are further away from each other as compared to the subsets from coupling strengths of the same network. The prediction of correlation matrices of different network structure is more difficult in the case of time series generated from ER networks.

The time series subset from SF networks created using the same oscillator clusters in a slightly different way. The low coupling strengths of the data

set clusters similar to the ER network by network structure while the high coupling strength of the data set clusters by coupling strength. The model can predict a missing coupling strength from the same network for all the coupling strengths used, it can predict for a different network structure for high coupling strength values with better accuracy than low coupling strength values.

Bibliography

[1] Austin Benson and Jon Kleinberg. Link prediction in networks with core-fringe data. In *The world wide web conference*, pages 94–104, 2019.

[2] Stephan Bialonski, Marie-Therese Horstmann, and Klaus Lehnertz. From brain to earth and climate systems: Small-world interaction networks or not? *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 20(1):013134, 2010.

[3] Victor M Eguiluz, Dante R Chialvo, Guillermo A Cecchi, Marwan Baliki, and A Vania Apkarian. Scale-free brain functional networks. *Physical review letters*, 94(1):018102, 2005.

[4] Huawei Fan, Junjie Jiang, Chun Zhang, Xingang Wang, and Ying-Cheng Lai. Long-term prediction of chaotic systems with machine learning. *Phys. Rev. Research*, 2:012080, Mar 2020.

[5] Huawei Fan, Junjie Jiang, Chun Zhang, Xingang Wang, and Ying-Cheng Lai. Long-term prediction of chaotic systems with machine learning. *Physical Review Research*, 2(1):012080, 2020.

[6] David Freedman, Robert Pisani, and Roger Purves. Statistics (international student edition). *Pisani, R. Purves, 4th edn. WW Norton & Company, New York*, 2007.

[7] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008.

[8] Amir Ghasemian, Homa Hosseinmardi, Aram Galstyan, Edoardo M Airoldi, and Aaron Clauset. Stacking models for nearly optimal link prediction in complex networks. *Proceedings of the National Academy of Sciences*, 117(38):23393–23400, 2020.

[9] Parham Ghorbanian, Subramanian Ramakrishnan, Alan Whitman, and Hashem Ashrafiuon. A phenomenological model of eeg based on the dynamics of a stochastic duffing-van der pol oscillator network. *Biomedical Signal Processing and Control*, 15:1–10, 2015.

[10] David Hallac, Youngsuk Park, Stephen Boyd, and Jure Leskovec. Network inference via the time-varying graphical lasso. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 205–213, 2017.

[11] Alan L Hodgkin and Andrew F Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of physiology*, 117(4):500, 1952.

[12] Radu Horaud, Florence Forbes, Manuel Yguel, Guillaume Dewaele, and Jian Zhang. Rigid and articulated point registration with expectation

conditional maximization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(3):587–602, 2010.

[13] Željko Kereta and Timo Klock. Estimating covariance and precision matrices along subspaces. *Electronic Journal of Statistics*, 15(1):554–588, 2021.

[14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[15] Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. Self-normalizing neural networks. In *Proceedings of the 31st international conference on neural information processing systems*, pages 972–981, 2017.

[16] Olga I Moskalenko, Alexey A Koronovskii, Alexander E Hramov, and Stefano Boccaletti. Generalized synchronization in mutually coupled oscillators and complex networks. *Physical Review E*, 86(3):036216, 2012.

[17] Louis M Pecora, Thomas L Carroll, and James F Heagy. Statistics for mathematical properties of maps between time series embeddings. *Physical Review E*, 52(4):3420, 1995.

[18] Michael G. Rosenblum, Arkady S. Pikovsky, and Jürgen Kurths. Phase synchronization of chaotic oscillators. *Phys. Rev. Lett.*, 76:1804–1807, Mar 1996.

[19] Brittany H Scheid, Arian Ashourvan, Jennifer Stiso, Kathryn A Davis, Fadi Mikhail, Fabio Pasqualetti, Brian Litt, and Danielle S Bassett.

Time-evolving controllability of effective connectivity networks during seizure progression. *Proceedings of the National Academy of Sciences*, 118(5), 2021.

[20] Jonathan Schiefer, Alexander Niederbühl, Volker Pernice, Carolin Lennartz, Jürgen Hennig, Pierre LeVan, and Stefan Rotter. From correlation to causation: Estimating effective connectivity from zero-lag covariances of brain signals. *PLoS computational biology*, 14(3):e1006056, 2018.

[21] Joel N Tenenbaum, Shlomo Havlin, and H Eugene Stanley. Earthquake networks based on similar activity patterns. *Physical Review E*, 86(4):046107, 2012.

[22] Muhan Zhang and Yixin Chen. Link prediction based on graph neural networks. *Advances in neural information processing systems*, 31, 2018.