Estimation Of Correlation From Limited Time-Series Data Using FitzHugh-Nagumo Model

Indian Institute Technology Indore

By

Prashant Singh Lohiya

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Science

Discipline of Physics

June, 2022

SUPERVISOR

Sarika Jalan

INDIAN INSTITUTE OF TECHNOLOGY INDORE

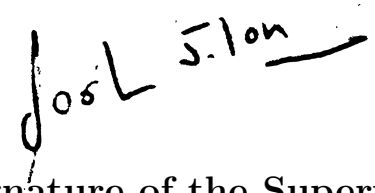
CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the thesis entitled **"Estimation Of Correlation From Limited Time-Series Data Using FitzHugh-Nagumo Model"** in the partial fulfillment of the requirements for the award of the degree of MASTER OF SCIENCE and submitted in the DISCIPLINE OF PHYSICS, **Indian Institute of Technology Indore**, is an authentic record of my own work carried out during the time period from August, 2021 to June, 2022 under the supervision of **Dr. Sarika Jalan**, Professor, Discipline of Physics, Indian Institute of Technology Indore. The matter presented in this thesis has not been submitted by me for the award of any other degree of this or any other institute.

Signature of student with date

Prashant Singh Lohiya

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

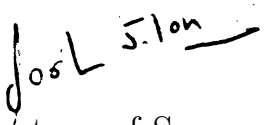


Signature of the Supervisor

of M.Sc. Thesis (with date)

Dr. SARIKA JALAN

PRASHANT SINGH LOHIYA has successfully given her M.Sc. Oral Examination held on 6/7 June, 2022.



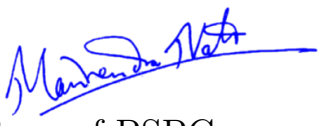
Signature of Supervisor

of M.Sc. thesis

Date: 27/05/2022

Convener, DPGC

Date:



Signature of PSPC

Dr. MANAVENDRA N. MAHATO

Date:

Signature of PSPC

Dr. SOMADITYA SEN

Date: 29/05/2022

ABSTRACT

Since the early 2000s, complex systems have been explored in depth and have been an important topic of research due to tremendous discoveries in real-world networks such as computer, biological, brain, climate, and social networks. Exploring a network and exploiting its dynamics to generate predictions is simple as long as we have information for all of the network's underlying nodes. But, what if this comprehensive knowledge about the network and its nodes is unavailable, as is the case with real work phenomena? Hereby, employing machine learning techniques, we offer a collective research of working just with a limited number of nodes in a network and using restricted time series of these few available nodes to predict correlation matrices. Feed Forward Neural Network is the machine learning algorithm we implemented in this study. Fitzhugh-Nagumo oscillators control the network's dynamics, and the coupled dynamics of these oscillators are utilised to generate synthetic data i.e. time series of the underlying nodes of the network.

ACKNOWLEDGEMENTS

To begin, I'd want to thank my mentor, Prof. Sarika Jalan, for welcoming me into the Complex Systems Lab and treating me as an integral member of the team. I am grateful for her unwavering support during the last year of my endeavour. Her strong understanding of the subject and zeal for research have been really beneficial. She taught me how to conduct great research in addition to the intricacies of the subject. Her dedication to producing relevant, innovative, and high-quality work has always been an inspiration to me, and it has aided me in pushing myself beyond my comfort zone to seek new and intriguing ideas throughout this project. I would also like to express my heartfelt gratiutde to my PSPC members, Dr. Manavendra Mahato and Dr. Somaditya Sen , for scrutinising my work and providing me with helpful suggestions on how I may improve both my work and my presentation, following their interactions with me.

I'd want to express my deepest appreciation to all of my seniors at the complex systems lab. You have been there for me through all of the thick and thin of this endeavour. I have learned a lot from everyone at CS lab. CSL seemed like a family to me because of my great seniors. Thank you so much for allowing me

to be a part of this family once more.

I'd want to make a particular mention to my batchmate Nikhil Easaw, who has been extremely supportive of me throughout. We've enjoyed a few laughs and memories together, and he's been a constant source of encouragement throughout this effort. Now, to the most important people in my life, I want to express my gratitude to my mother and father for bringing me into this world and for providing me with a comfortable and resourceful existence that has allowed me to follow my passions. None of this, or any other fantastic accomplishment in my life, would have been possible without you. I'd also like to express my gratitude to my sister for providing me with all of the assistance I've needed throughout my life.

Contents

List of Figures

Chapter 1

Introduction

Networks are to be found EVERYWHERE! Whether we talk about our social groups, our working-areas or most importantly in nature like biological areas and climate. A network is a system or a collection of certain entities and is characterised by the type of interaction/connection between those entities. A complex network, in of network theory, is a graph containing certain additional significant traits that are not clearly traceable in simple networks such as lattices but are regularly found in networks reflecting actual real-world systems.. Since the 2000s, the study of complex systems has emerged as a young and hot-spot area of scientific research, partly motivated by the pragmatic results of real-world networks such as computer, biological, brain, cli-

mate, and social networks, among others..The problems related to complex systems are in abundance around us, and such systems are deliberated and fathomed by constructing networks. The dynamical behaviour of a node is either observed to be in a synchronised or in a desynchronised state i.e. either all the nodes will have the same behaviour or the nodes will act independently from each other.

Neurons are the basic information processing units of a living being that generate electrical impulses, known as action potentials, in response to chemical and other inputs and send them to neighbouring cells via synaptic connections (points where two neurons link). As a result, neurons interact directly via ion diffusion and indirectly via their shared environment. Neurons do not exist in isolation. They form intricate networks that cover the entire body, and communication between neurons is accomplished through travelling electrical signals known as action potentials, which are determined by the cell's electrophysiological properties. Typical neurons can make up to 10,000 synaptic contacts with neighbouring neurons and up to 100,000 synaptic contacts on rare occasions [13]. Understanding the linked behaviour of the neural system is therefore crucial from both an academic and practical standpoint. Excitability is a characteristic shared

by many physical and biological systems[5]. Since the pioneering work of Hodgkin and Huxley and the emergence of the basic mathematical models by various authors reported research on the subject has grown enormously. A single neuron, for example, is excitable in the sense that a tiny disturbance from its immobile state, i.e. a steady stationery value of the cross membrane potential, can result in a considerable surge of its potential before reverting to its inactive state. Aside from neurons, several other cells are known to generate potential spikes across their membrane. Excitable units are often hardwired parts of complex systems that may transmit excitation between them.

Machine learning (ML) is a fundamental subfield of artificial intelligence (AI). Machine learning applications tend to be accurate by learning from the experience like the humans do without direct programming. When unveiled to new data, these ML applications learn, grow, change and evolve by themselves. In other words of explanation, machine learning learning involves computers detecting intuitive information from the data without actually being told where to look. Instead, ML application do this by leveraging the power of algorithms that learn from the data in an iterative process.Machine learning techniques have been applied in diverse areas for predicting and understanding

the behaviour of complex systems and also in understanding the important parameters that aid effective predictions. In real world phenomenons, it is often observed that the limited time series of a dynamical process is available only for a small number of nodes of the system.

Due to availability of information of only a few number of nodes present of the system, predicting any results from such a limited amount of data becomes a tedious job.There are various applications of constructing the correlation matrix of the time series of nodes, for example, 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 by setting a threshold value for the correlation[3]. Further more this simple method of link prediction using cross correlation of time series data has been used to predict the link between two nodes based on the strength of their cross correlation and then applied to various real world problems like prediction of Earthquakes[19]. The correlation between fMRI is used to extract functional states connecting correlated human brain sites[17][7].These 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.

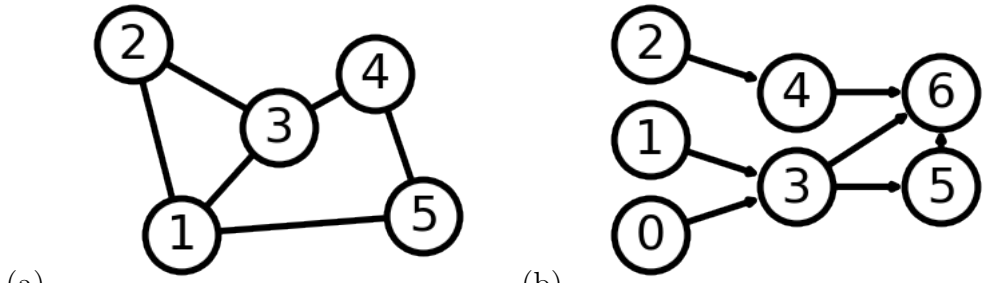
There are a number of methods for estimating the true covariance matrices like the MLE, Maximum Likelihood Estimation and the GLASSO, Graphical Least Absolute Shrinkage and Selection Operator Method[9], and the evolved versions of GLASSO such as P-GLASSO and DP-GLASSO[11] but the prerequisite of all these methods is that they require the information of all the nodes present in the system, but as we look upon the real world phenomenons this complete information is rarely available and distinguishing between the most connected nodes is a difficult task. As a result, these approaches flounder to depict scenarios where the size of the network is significantly more than the number of time series available. And thus, we aim our study at investigating a machine learning-based strategy for reconstructing the correlation matrix of the complete network using the limited time series of a few nodes.

Chapter 2

Model and Techniques

2.1 Network

To have a comprehensive understanding of a complex system, we need to perceive how the underlying components of the network interact with each other. A system might be made up of various kinds of networks and that network consists of the nodes and these nodes are connected to each other via edges. N represents the number of nodes and it is often referred to as the size of the network.



(a) (b)

Figure 2.1: (a) Undirected Graph consisting of $5-nodes$ and $6-edges$, (b) Directed Graph consisting of $7-nodes$ and $6-edges$.

2.1.1 Degree

The degree of a network is measured by the number of connections a node has with every other node present in the network. It is denoted by k, the degree of the ith node is k_i

2.1.2 Average Degree

The average degree of a network is denoted by $< k >$ and is given as :

$$< k >= \frac{\sum_{i=1}^{N} k_i}{N} \tag{2.1}$$

2.2 Time series Data

Time series data is a sequence of data points catalogued in chronological order. These data points are used to track change over time and generally consist of sequential measurements taken from the same source across a time interval. Time series data is abundant and present everywhere in our technological era, because time is an integral part of every observable quantity. As the world is increasingly getting instrumented, sensors and systems are constantly venting unabated stream of time series data. Such enormous of data has applications in various fields and industries, some of the examples are :

- Electrical activity in the brain
- Rainfall measurements
- Stock prices
- Annual retail sales
- Monthly subscribers
- Heartbeats per minute

2.3 Correlation Matrices

For describing a relationship between M variables, we opt to use correlation matrices. The diagonal elements of an MxM square symmetrical matrix correlation matrix, are always equal to 1 owing to the fact that they self-correlated and off-diagonal ij_{th} elements equal to the correlation coefficient between the i_{th} and j_{th} variables

2.4 Theoretical Model

It was in 1952 that Hodgkin and Huxley[10] constructed a model of excitable nerve membrane on the basis of a series of experiments on a giant squid axon. In this landmark study, a four-dimensional (V, m, h, n) model of ionic processes governing current conduction and action potential excitation in nerve was developed, ushering in a new age of electrophysiological investigations. The Hodgkin and Huxley equations (abbreviated HH equations) were initially developed under voltage-clamp conditions[16]. However, numerical solutions to the equations have been obtained under various situations. Hodgkin and Huxley (1952) determined the equations' responses to an abrupt shift in the

membrane potential. Under space-clamp conditions, Cole et al. calculated the membrane potential to step-current stimuli. Derived from the pioneer works of Hodgkin and Huxley, R.FitzHugh(1961) examined the excitation process of the Hodgkin-Huxley equations to a sudden displacement of membrane potential using the phase-plane analysis and the following next year, J. Nagumo developed the analogous circuit, describing a sample model of an excitable system (e.g., a neuron).The FHN model's enormous success is owed not only to its clear mathematical structure and affluence in system dynamics, but also to its correspondance to the Hodgkin-Huxley (HH) model..

In actuality, the four-dimensional HH model may be subdivided into two parts: (V, m) represents v in the FHN model, with rapid dynamics reflecting excitability and (h, n) represents w, with slow dynamics representing accommodation and refractoriness[22]. As a consequence of these properties present in the FHN model can be approximated as other ionic mod-

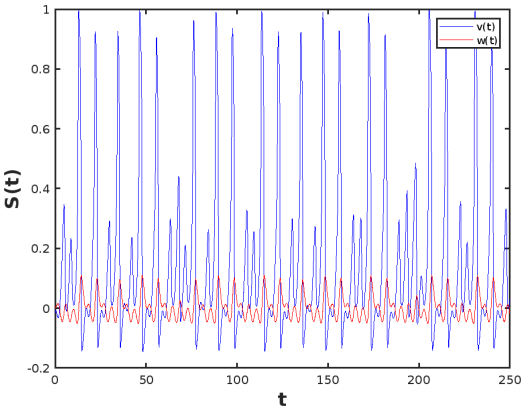


Figure 2.2: Chaotic time series of FitzHugh-Nagumo oscillator for $a = 0.42$.

els. These certainities make FHN model one of the most studied models in analytical neuroscience/electrocardiology[8][15]. Around six decades have passed since the FitzHugh-Nagumo model was initially published, and a variety of variants of the original model have been produced since then. The following two-dimensional, nonlinear ordinary differential equations[18] represent the dynamical system of a single, solitary neuron:

$$\begin{aligned} \dot{v} &= \frac{1}{\delta}[v(v-a)(1-v)-w] \\ \dot{w} &= v - w - b + S(t) \end{aligned} \tag{2.2}$$

where, v is the membrane potential, w is the recovery variable and S(t) is the sinusoidal driving signal: S(t)=$rsin\omega_o t$. The other parameters are $a = 0.42$, $b = 0.15$, $\delta = 0.005$, $r = 0.2$, $\omega_o = 15.0$.

2.5 Pearson Correlation

The world is a loud place from the standpoint of signalling. We must be careful with our attention in order to make sense of anything. We as humans have gotten rather proficient at filtering

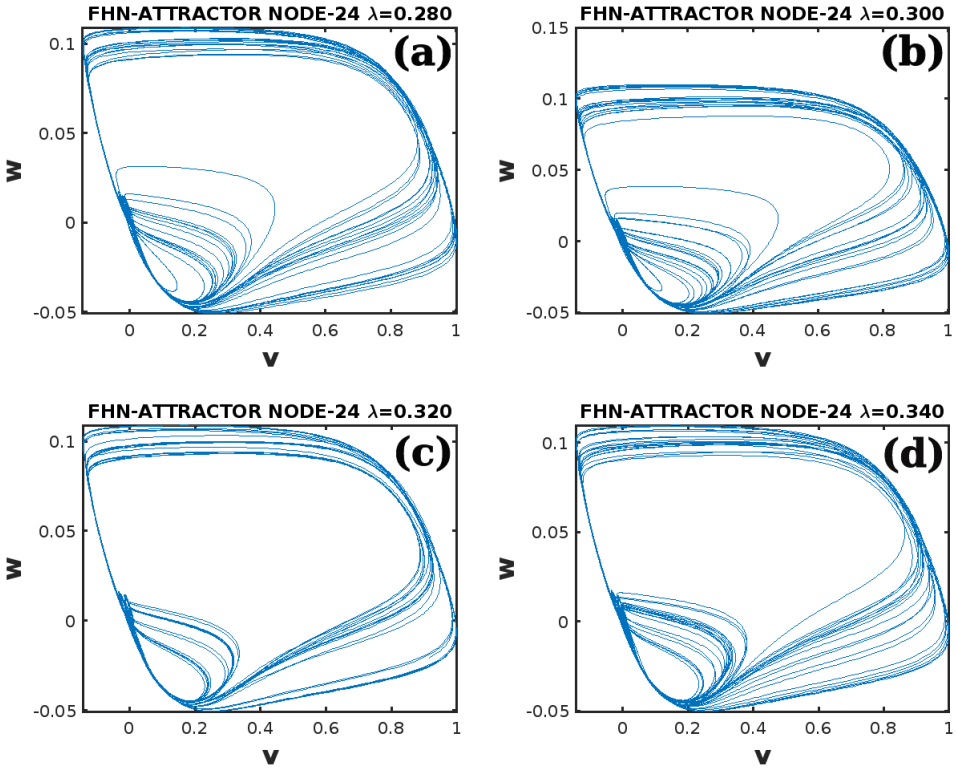


Figure 2.3: Phase portrait of FitzHugh-Nagumo oscillators for different coupling strengths, (a) $\lambda = 0.280$, (b) $\lambda = 0.300$, (c) $\lambda = 0.320$, (d) $= 0.340$

out background signals over millions of years of natural selection. We have developed the ability to link certain signals to specific occurrences. For instance, if someone is playing cricket and they are trying to a catch the ball. Now, your brain must repeatedly sample the ball's present position and estimate its future trajectory in order to predict its motion. Advanced players will additionally consider any spin provided to the shot by their opponent. All of this necessitates a significant amount of implicit differential calculus.We take it for granted that our nervous system can do this automatically in most cases.

This may seem so self-evident as to be unworthy of mentioning, yet it demonstrates how adept humans are at learning to generate correct predictions from noisy evidence. Given a constant stream of audiovisual input, a blank-state machine would undoubtedly struggle to figure out which signals best indicate the appropriate course of action. Fortunately, statistical and computational approaches exist for detecting patterns in complicated data.

When we talk about the "correlation" of two variables, we generally mean their "relatedness" in some sense. Correlated variables share information about one another.The stronger the link between the variables, the more information one variable may

provide about the other. "*Correlation does not imply causation*". This is unquestionably true, there are numerous reasons why even a large correlation between two variables does not imply causation. The apparent association might be attributable to the impacts of a third variable that isn't visible, or it could simply be due to chance.However, correlation allows one to make predictions about one variable based on another.For both linear and non-linear data, there are numerous approaches for estimating correlated-ness, one of which we have considered[4].

We have used the most widely considered correlation coefficient i.e. the Pearson's Correlation Coefficient or Pearson's r. Mathematically, it is defined as "the covariance of two vectors, normalised by the product of their standard deviations". It is given as :

$$r = \frac{\sum(v_i - \bar{v})(w_i - \bar{w})}{\sqrt{\sum(v_i - \bar{v})^2(w_i - \bar{w})^2}} \tag{2.3}$$

Chapter 3

Methodology

We manoeuvre a machine learning technique to predict the cross-correlation matrix for the whole network given a subset of time series for a few nodes. We take a network with N number of nodes and m time steps of the used to calculate the true cross-correlation matrix $\mathbf{R}$ of the time series of the network. Then the entire time series information of the dynamical system can be expressed by a matrix $\mathbf{T}$ where T_{ij} represents the time-series information of the i^{th} node at the j^{th} time step. The corresponding Pearson's r, between the time series of nodes given by a matrix $\mathbf{R}$, where R_{ij} represents the correlation between the time series of i^{th} node and j^{th} node. We predict the entire correlation ma-

trix from a subset $\delta\mathbf{T}$ of the original time series $\mathbf{T}$

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

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 $n < N$.

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

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

we compare the predicted correlation matrix $\widetilde{R}$ to the true correlation matrix $\mathbf{R}$ ($\widetilde{R} \approx \mathbf{R}$) and use mean square error as a measure of similarity between the predicted and true correlation matrices. The correlation used is the linear the Pearson correlation coefficient r (same as section 3) where r is given as

$$r = \frac{\sum(v_i - \bar{v})(w_i - \bar{w})}{\sqrt{\sum(v_i - \bar{v})^2(w_i - \bar{w})^2}} \tag{3.1}$$

with x_i and y_i representing the time evolution data of each node after the initial transient, and t_{ij} is the time evolution of a node at each time step. The number of input time steps k is 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.

3.1 Synthetic Data Generation

The dynamical system is modelled using two most established random graph models, the Erdos-Renyi (ER) and Scale-Free (SF) networks. The system's time series is generated by the coupled dynamics of the nodes on these graph models.

3.1.1 Erdos-Renyi Network

Erdos-Renyi (ER) network by definition can be explained as $G(N, p)$ being a random graph with N number of nodes where each possible edge has probability $p = N/\langle k \rangle$ of existing[2]. ER random graphs do not follow the power law of degree distribution. The number of edges in $G(N, p)$ graph is random variable

with expected value $\begin{bmatrix} N \\ 2 \end{bmatrix} p$.

3.1.2 Scale-Free Network

Debating about the real-world networks they are an outcome of a phenomenon called growth and preferential attachment that continuously increases the number of nodes and is thereby contrary to the random graphs, as they assume the number of nodes to be fixed. New nodes in a real-world network prefer to link to more connected nodes, but nodes in random networks pick their interaction partners haphazardly. Recognizing that growth and preferred attachment coexist in actual networks resulted in the Barabasi-Albert (BA) model, which is used to build Scale-Free networks[2]. In the BA or Scale-Free model, we begin with *no* nodes, the ties between which are chosen at random, given that one node has at least one edge.e. The network develops in the following steps:

- Growth : A new node $\mathrm{n}_o(\leq n)$ is added at each time-step.
- Preferential Attachment : The degree ki determines the likelihood $pi(\mathrm{k})$ that a link of the new node links to node i:

$$\pi(k_i) = \frac{k_i}{\sum_j k_j} \tag{3.2}$$

The time evolution of the phase of each node is geared by the **v** coordinate of the FitzHugh-Nagumo oscillator in phase space whose equation is given by the following equations[14] :

$$\begin{aligned} \dot{v_i} &= \frac{1}{\delta}[v_i(v_i - a)(1 - v_i) - w_i] + \sum_{i=1}^{N} \lambda_{ij}(v_j - v_i) \\ \dot{w_i} &= v_i - w_i - b + S(t) \end{aligned} \tag{3.3}$$

where , $i = 1, \ldots, N$, v is the membrane potential, w is the recovery variable, λ_{ij} is the coupling strength between the i^{th} and the j^{th} neurons (nodes), $S(t)$: S(t)=rsinωt. The frequency of the driving signal has been derived from a normal degree distribution having mean 1 and variance 0.03 this frequency is given by $\omega+ \Delta\omega$ here the $\Delta\omega = 0.001k$ where k is generated from a normal distribution. The frequency is the mismatched parameter in the set of oscillator and λ denotes the overall coupling strength between the connected nodes.The other parameters of the oscillators are : a = 0.42 and the other parameters are b=0.15, δ =0.005, r=0.2, ω=15.0.

3.2 Global Synchronisation Error

The Global Synchronization Error (ΔE) of the time series of the nodes over the measurement window is used to determine the system's level of synchronisation. The following formula is used to determine the (ΔE) of the time series matrix $\mathbf{T}$ of order $m \times N$:

$$\Delta E_i = \frac{1}{m(N-1)} \sum_{j=1}^{N} \sum_{t}^{t+m} ||v_j - v_i|| \tag{3.4}$$

The absolute difference between the phase of each oscillator corresponding to a node is computed with every other nodes for the full length of the time series, for each phase space coordinates v, w gives the error of the i^{th} node where $||v||$ represents $\sqrt{v^2 + w^2}$ where v,w are phase space coordinates of the FitzHugh-Nagumo oscillator. This procedure is followed for all N nodes and their average value gives the Global Synchronization Error of the time series for a particular coupling strength (λ). The transitioning of the system of oscillators towards the synchronisation is depicted by the ΔE vs. λ curve. We want to locate a semi-synchronous zone where the nodes are neither entirely unsynchronized nor completely synchronised, i.e. a region where neither strongly correlated nodes nor weak correlations exist.

The semi-synchronous region of the dynamical system is determined from the coupling strengths corresponding to the middle portion of the E vs λ plot in Fig.4. For the FitzHugh-Nagumo oscillator, this corresponds to the region with E lying between 0.16-0.18 for Erdos-Renyi network and 0.11-0.13 for Scale-Free network. After leaving a considerate initial transient, the time series (phase development of the nodes with time) is recorded for a few values of such coupling strengths. To represent the time series of one of the dynamical system's nodes, we only utilise one of the two phase space variables, over here we have used variable v of eq.(3.2).

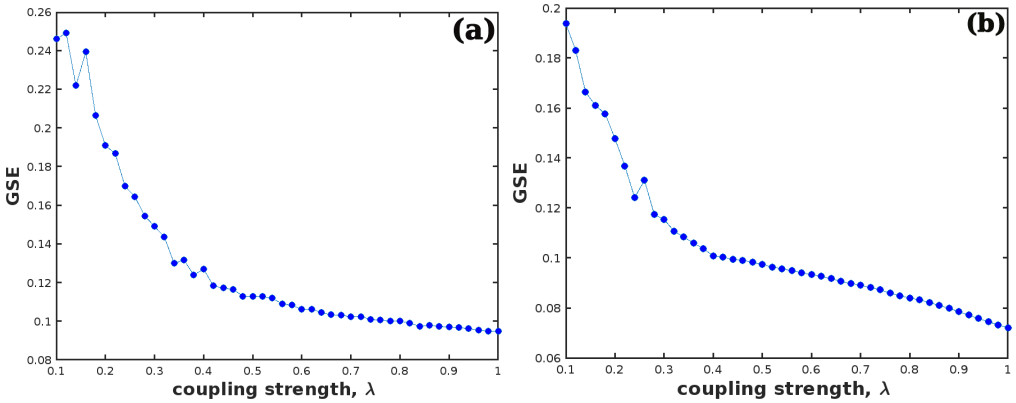


Figure 3.1: Global Synchronisation Error (GSE) vs. λ (a) GSE for ER-network with N=100, $< k >= 10$, (b) GSE for SF-network with N=100, $< k >= 10$

3.3 Machine Learning Algorithm

Artificial neural networks (ANNs) are the networks of basic processing units (referred to as "neurons") that operate on local data while connecting with other elements. The structure of a real brain inspired the construction of ANNs, however the processing components and architectures employed in ANNs have diverged significantly from their biological inspiration.

Although, due to the technological advancements there are many different forms of neural networks that exist today, but the fundamental concepts are relatively similar. Each neuron in the network has the ability to accept, analyse, and transmit input signals. No neuron in the network is is isolated, i.e. each neuron is linked to at least one other neuron, and the significance of that connection in the neural network is determined by assigning a weight coefficient to each connection present in the network. In theory, a neural network has the power of being compared to a *universal approximator*[12] , elucidating to the fact that it can perform any arbitrary mapping from one vector space to another. The major benefit of neural networks is that they may make use of previously undiscovered information concealed in data (but they are not able to unsheathe it). The process of

capturing this hidden information or the unknown facts within the data is the *Learning of neural networks* or *Training of neural networks*. In mathematical formalism, learning entails adjusting the weight coefficients such that certain requirements are met for better applications of neural networks.
Mainly there are two types of training processes which exist and they are as follows :

- **Supervised Learning** : As the name implies, supervised learning entails working under supervision. Supervised learning, as the name indicates, entails the presence of a supervisor who simultaneously functions as a teacher. In essence, supervised learning is the use of well-labeled data to guide or train a computer. This indicates that some information has already been tagged with the right answer. The computer is then given a fresh collection of examples (data), which the supervised learning algorithm analyses and delivers a proper result from labelled data. Some forms of Supervised learning are :-
  - Regression
  - Logistic Regression
  - Classification

  - Naive Bayes Classifiers
  - K-NN (k nearest neighbors)
  - Decision Trees
  - Support Vector Machine

- **Unsupervised Learning** : Unsupervised learning is the process of teaching a machine to operate on data that hasn't been classed or labelled and letting the algorithm to act on it without supervision. Without any prior data training, the machine's mission here is to sort unsorted data into categories based on similarities, patterns, and differences. Unlike supervised learning, no teacher is present, which implies that the machine will not be trained. As a result, the machine is left on its ability to discover hidden structure in unlabeled data on its own.
  - Clustering
  - Association
  - Dimensionality Reduction

3.3.1 Feed Forward Neural Network

In our study, for predicting correlation matrix from the time series of a few nodes, we employ a Feed-Forward Neural Network (FNN). Our challenge involves supervised learning using time series inputs from a few nodes and correlation matrix outputs.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. **The neural network's input is the samllest subset of the time series data corresponding to the nodes, and the network's output is the predicted correlation matrix.** Except for the output layer, we use the SELU (Scaled Exponential Linear Unit) as a fundamental activation function. We choose the sigmoid function[6] for the output layer since the correlation value lies between 0 and 1. For neural network optimization, we employ Adam (Adaptive Moment Estimation method). A multi-layer perceptron (MLP) is another term for a feed-forward neural network[21]. A model was built with one input layer, two hidden layers, and one output layer. A layer is made up of numerous neurons, and neurons in neighbouring layers are linked together. The relationship between the input($a_j^{(l-1)}$) and output($a_i^{(l)}$) of a layer is as follows:

$$z_i^{(\ell)} = \sum_{j=1}^{K} w_{j,i}^{(\ell-1)} a_j^{(\ell-1)}, \tag{3.5}$$

$$a_i^{(\ell)} = \mathrm{SELU}(z_i^{(\ell)}). \tag{3.6}$$

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[14]. 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.

$$Loss = -\sum_{k=1}^{N_L} (Y_k log(a_k^{(L)}) + (1 - Y_k) log(1 - a_k^{(L)})), \tag{3.7}$$

We use Adam algorithm[14] for the minimizing the loss function. Feedforward networks (FFN), Convolution neural networks (CNN), and Recurrent neural networks (RNN) are the three major types of neural networks. RNNs are effective for time series predic-

tion. It is, for example, appropriate for the task of forecasting t_n given data ranging from t_0 to $t_(n-1)$. As a result, it is beneficial for learning consecutive facts. CNNs are also effective for learning 2D and 3D data characteristics. For example, consider the difficulty of locating a certain object in a photograph or identifying photographs of dogs and cats. Because CNN is extensively used for learning 2D and 3D form of data, it is appropriate for our challenge. However, the random network data that we use is difficult to characterise neighbouring nodes, unlike the image, thus CNN was not an appropriate choice either. CNN performs well when nearby pixels have a connection, such as in a photograph.Finally, the Feed-Forward Neural Network, also known as the Fully-connected Network (FCN). FCN, as the name implies, is linked to all nodes. As a result, it was suitable for creating graph-type data such as random networks. There are also Graph Neural Networks (GNN) that learn graph-type data effectively. However, because our aim is to learn and predict data that represents the graph's features, GNN, which are used to learn the graph itself, were not appropriate for our application.

3.3.2 UMAP

We aim to explain why a machine learning algorithm may predict correlation matrices using an unsupervised learning technique called UMAP (Uniform Manifold Approximation and Projection)[1], which demonstrates the specific qualities of time series data that can be learnt so that the neural network can gain something useful. Dimensionality reduction can help machine learning practitioners view and understand huge, high-dimensional datasets. t-SNE[20] is a popular visualisation approach, although its performance decreases with large datasets, and utilising it effectively may be difficult. UMAPs help to comprehend the inherent qualities of the data set. A subset of the time series corresponding to all the coupling strengths used to create the data set is fed into the UMAP algorithm for one time window. UMAP then converts the time series column vector used for prediction into points in a two-dimensional space. The difference between the various subsets utilised for prediction is represented by the distance between the points in this 2-Dimensional space. Only the points which have similar properties, make up a cluster, a collection of points that are part of such clusters can be extrapolated from it even if they aren't in the training dataset.

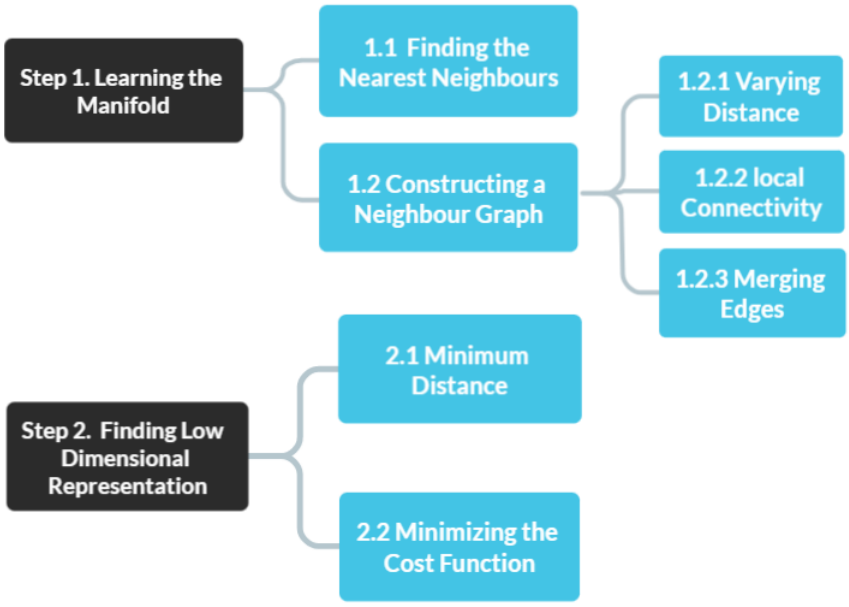


Figure 3.2: Components and Steps of UMAP in brief.

Chapter 4

Results and Discussion

The results presented in this study are for the correlation matrix predicted using synthetic time series data generated through the coupled dynamics FitzHugh-Nagumo oscillator on Erdos-Renyi and Scale-Free networks. Feed Forward Neural Network is The correlation matrix is predicted using synthetic time series data created by coupled FitzHugh-Nagumo oscillator networks, and the findings of this study are shown in this section. The upper triangular section of the correlation matrix is predicted using the machine learning technique-Feed forward neural network. Lower triangular elements are determined by equating to the equivalent upper triangular elements by symmetry using the formula $\widetilde{\boldsymbol{R}_{ij}} = \widetilde{\boldsymbol{R}_{ji}}$, and diagonal elements of the projected correlation

matrix $\widetilde{\boldsymbol{R}}$ are equated to 1 as they are self correlated. The symmetry in the correlation matrix is then used to derive the whole correlation matrix. By altering the training and testing data, the model's ability to predict the correlation matrix is proven.The various types of training and testing data are discussed in depth below. We have considerable information of how well the model performs since the complexity of the examples investigated rises systematically when the limitations in the training data are reduced. The case that we attempt to predict a correlation matrix for is when the network realization is kept the same and the coupling strength is varied. A network of size $N = 100$ with average degree $< k >= 10$ was considered, both for Erdos-Renyi and Scale-Free networks respectively.

- **Training and Testing Data :** The training data for altering coupling strengths(λ) consists of restricted time series information $\delta\mathbf{T}$ of the number of nodes utilised for predicting the correlation matrix as input and their matching real correlation matrix $\mathbf{R}$ as training output. To generate the $\delta\mathbf{T}$ for variable coupling strengths(λ) with their genuine correlation matrix $\mathbf{R}$, 75 distinct network architectures with the same average degree were employed. In order to min-

imise over-fitting, the coupling strength used to evaluate the prediction is not included in the training data for all 75 network configurations. the training data consists of the time series of a few nodes with different coupling strengths, say $\lambda_1, \lambda_2, \ldots, \lambda_{k-1}, \lambda_{k+1}, \ldots, \lambda_l$ for a network realization we predict the correlation matrix for λ_k in this particular case.

Now, it becomes very important to comprehend how well our technique works and how well are the predictions made by it. Thus, to determine the accuracy of the prediction, we gradually increase the number of nodes employed in the correlation matrix prediction, this is done for both ER and SF networks. We calculated the Mean Square Error (MSE), between the true correlation matrix $\mathbf{R}$ and the predicted correlation matrix $\delta\mathbf{R}$ to measure the accuracy of the predictions. It's crucial to understand not just the accuracy of the predictions, but also the threshold value of the minimum nodes necessary to create a decent prediction. For serving this purpose we formulate a performance array matrix, which shows convergence of the MSE of the predicted correlation matrix with the number of nodes and by varying the length of the time series.

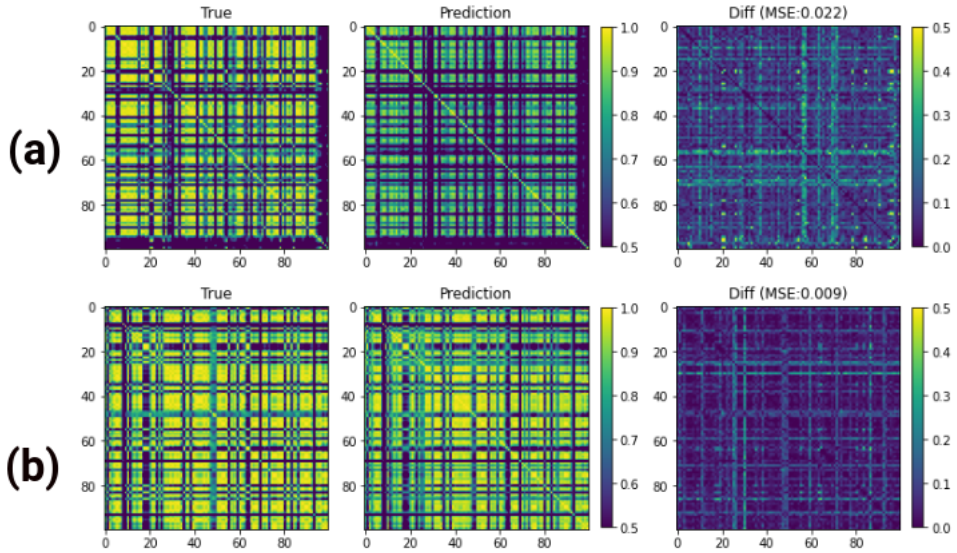


Figure 4.1: (a)The True and predicted correlation matrix for $N = 100$, $\langle k \rangle = 10$ network displayed for ER network for $\lambda = 0.320$, (b) The True and predicted correlation matrix for $N = 100$, $\langle k \rangle = 10$ network displayed for SF network for $\lambda = 0.280$

4.1 Predictions For ER-Random Network

We used NetworkX library for generating 75 different realisations of the same network. The erdos-renyi-graph graph generator function of NetworkX library returns a graph, $G_(N, p)$ which has N number of nodes, for our case the network size is $N = 100$ and p is the probability of possible edges and the average degree of the network is $< k >= 10$ for every realisation. The time series of the nodes was created by using the dynamics

of FitzHugh-Nagumo oscillators and the coupling strengths λ used for creating time series and the true correlation matrices are $\lambda = 0.280$, $\lambda = 0.300$, $\lambda = 0.320$, $\lambda = 0.340$. The machine is trained for coupling strengths $\lambda = 0.280$, $\lambda = 0.300$, $\lambda = 0.340$ and tested for $\lambda = 0.320$. Figure 4.1 (a) shows the true correlation matrix, the predicted correlated matrix and MSE matrix as measure of how close is the prediction to the true value.Figure 4.2 (a) illustrates the performance array matrix, which shows the convergence of the MSE of the predicted correlation matrix with the increase in the number of nodes.

Next, we bid to determine the minimum number of nodes required for a good prediction as well as we also try to focus onto limiting the length of the time series being used for the prediction. Figure 4.3 (a) depicts saturation point, threshold value for the minimum number of nodes required corresponding to the limited time series for a decent prediction to be made by the machine. The time series generated using FHN oscillators for the ER-Random network shows the clustering using UMAPs. As shown in figure(a) and figure(c), even though the different network realisations form separate clusters, yet they lie very close to each other, which is a consequence of the fact that the time series generated by FitzHugh-Nagumo Oscillators have

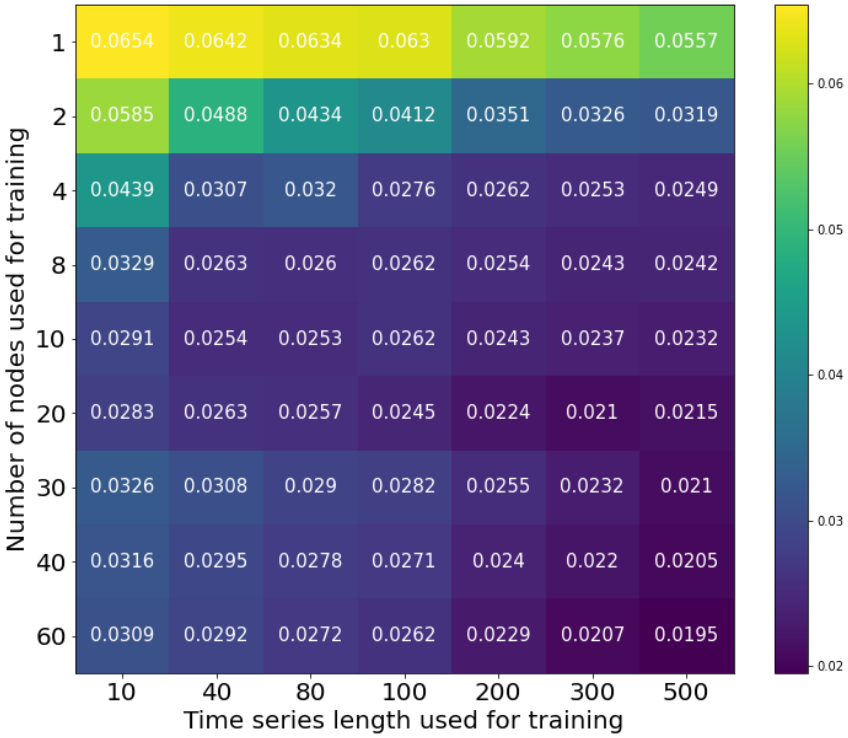


Figure 4.2: Performance array matrix depicting accuracy of the prediction with increasing number of nodes and length of limited time series for SF-network with $N = 100$ and $< k >= 10$.

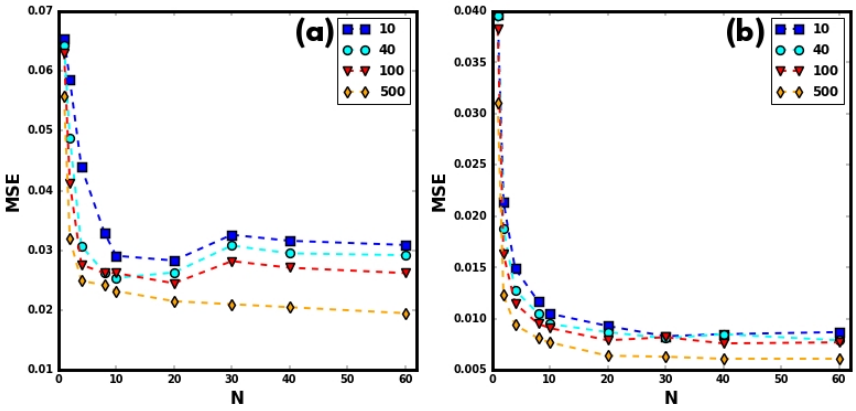


Figure 4.3: MSE vs. NN (Number of Nodes) plot showing the saturation with increase in the number of nodes with corresponding time series for (a) ER-network $N = 100$ and $< k >= 10$, (b) SF-network $N = 100$ and $< k >= 10$.

very similar properties. Similarly, in figure(b) and figure(d) the clustering is formed using UMAPs corresponding to the various coupling strengths(λ) used for creating the time series data set.Further, figures (e), (f), (g), and (h) show the UMAP clustering predictions for the correlation matrices calculated using all the network realisations and all the coupling strengths(λ).

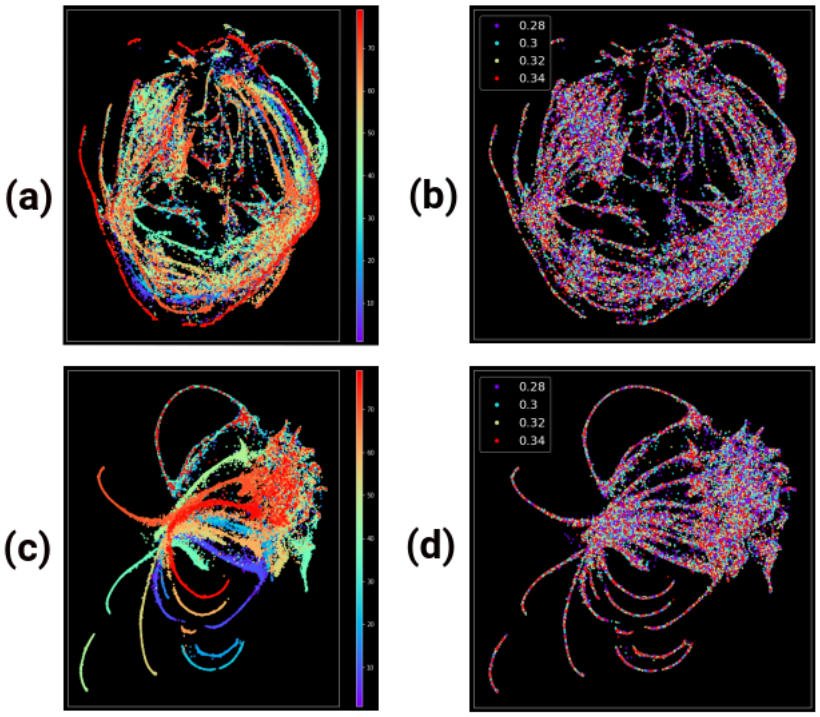


Figure 4.4: Results of dimensionality reduction using UMAP (a)UMAP labelled by network realisations for $nodes = 10$ and $length\ of\ time\ series = 100$, (b)UMAP labelled by coupling strenths(λ) for $nodes = 10$ and $length\ of\ time\ series = 100$, (c)UMAP labelled by network realisations for $nodes = 40$ and $length\ of\ time\ series = 100$, (d)UMAP labelled by coupling strenths(λ) for $nodes = 40$ and $length\ of\ time\ series = 100$

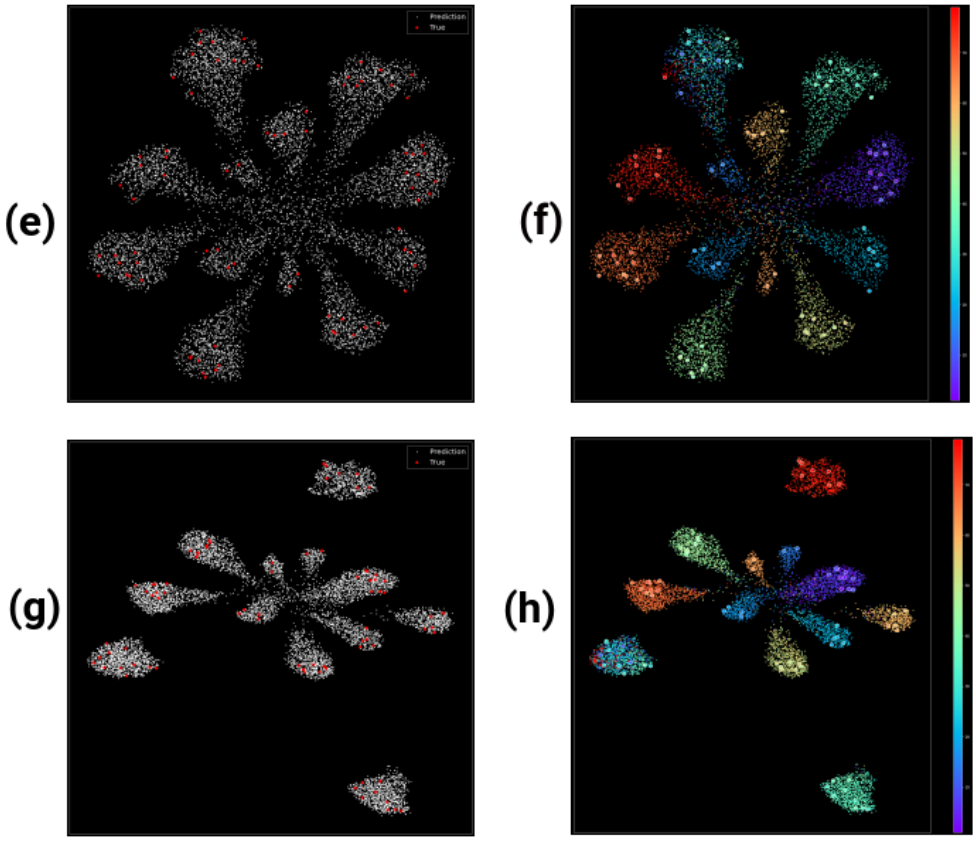


Figure 4.5: Results of dimensionality reduction using UMAP $nodes = 10$ and $lengthoftimeseries = 100$ (e) Predicted(white) and True(red) correlation matrices for $nodes = 10$ and $length\ of\ time\ series = 100$ and, (f)Prediction of correlation matrices labelled by network realisations for $nodes = 10$ and $length\ of\ time\ series = 100$, (g) Predicted(white) and True(red) correlation matrices for $nodes = 40$ and $length\ of\ time\ series = 100$ and, (h)Prediction of correlation matrices labelled by network realisations for $nodes = 40$ and $length\ of\ time\ series = 100$

4.2 Predictions For SF Network

Scale-Free network for the prediction has been created using NetworkX library's barabasi-albert-graph function, which returns a graph with N nodes with m being the number of edges to be connected from the new node to existing nodes. 75 different network realisations with $N = 100$ and $< k >= 10$ for each one of them. FHN oscillators were used to create the time series for the coupling strength λ used for creating time series and the true correlation matrices are $\lambda = 0.240$, $\lambda = 0.260$, $\lambda = 0.280$, $\lambda = 0.300$. The machine is trained for coupling strengths $\lambda = 0.240$, $\lambda = 0.260$, $\lambda = 0.300$ and tested for $\lambda = 0.280$. Figure 4.1 (b) shows the true correlation matrix, the predicted correlated matrix and MSE matrix as measure of how close is the prediction to the true value. Figure 4.3 (b) depicts saturation point, threshold value for the minimum number of nodes required corresponding to the limited time series for a decent prediction to be made by the machine. Figure 4.6 is the illustration of the performance array matrix, which shows the convergence of the MSE of the predicted correlation matrix with the increase in the number of nodes.

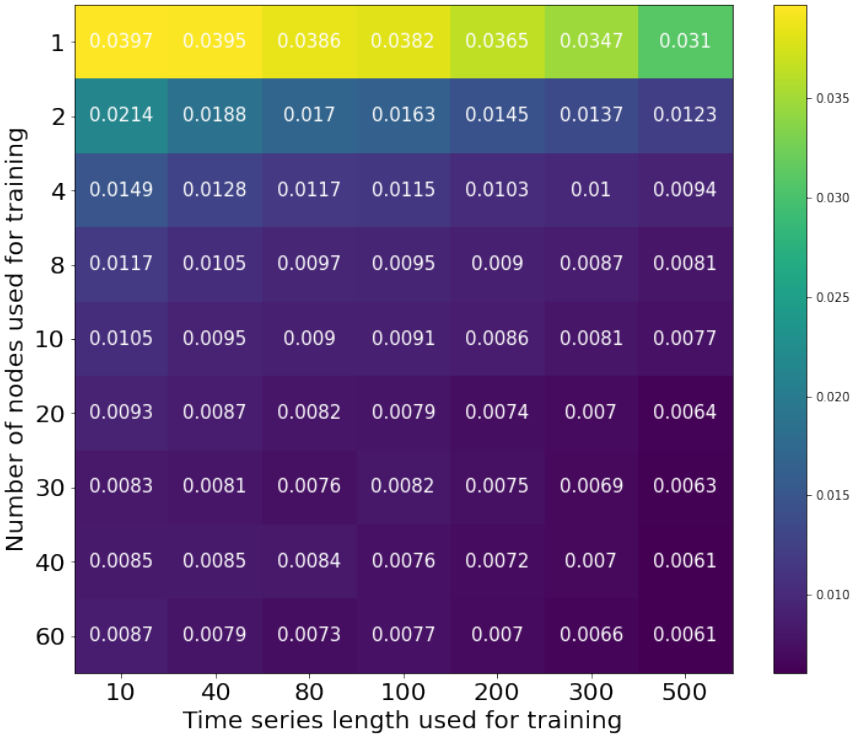


Figure 4.6: Performance array matrix depicting accuracy of the prediction with increasing number of nodes and length of limited time series for SF-network with $N = 100$ and $< k >= 10$.

Chapter 5

Conclusion

We worked on a really novel and distinct topic in this study thesis. Each technique has been reduced in a step-by-step manner. The paper demonstrates the use of a machine learning technique called Feed Forward Neural Network (FFNN) to predict correlation matrices using restricted time series of a few network nodes, in the following research we used exploited Erdós-Renyi and Scale-Free Networks, to illustrate the problem. The network dynamics, i.e. the creation of time series data for each node in the network, were performed using FitzHugh-Nagumo oscillators.

FitzHugh-Nagumo oscillators have very definitely never been employed for the production of network correlation matrices in

conjunction with the ML technique, making our approach highly novel.

Machine-Learning algorithm predictions produced by FFNN were validated by comparing the Mean Square Error (MSE) between the true correlation matrix and the predicted correlation matrix. The calculation of the threshold value for the number of nodes and the length of time series necessary to produce effective predictions has also been demonstrated by showing a graph between MSE and the number of nodes utilised for prediction with increasing time series.

UMAPS (Uniform Mani- fold Approximation and Projection), an unsupervised learning approach has also been used for determining similarity in time series data obtained on the basis of network realisations and coupling strengths(λ). Time series and correlations with comparable features are shown as single coloured clusters. Finally, the work given in this study demonstrates the capability of utilising our model to predict correlation matrices not only for synthetic data but also for real-world data such as stock market, EEG data, and so on.

Bibliography

[1] Mebarka Allaoui, Mohammed Lamine Kherfi, and Abdelhakim Cheriet. Considerably improving clustering algorithms using umap dimensionality reduction technique: a comparative study. In *International Conference on Image and Signal Processing*, pages 317–325. Springer, 2020.

[2] Albert-László Barabási. Network science book. *Network Science*, 625, 2014.

[3] 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.

[4] J Martin Bland and Douglas G Altman. Correlation, regression, and repeated data. *BMJ: British Medical Journal*, 308(6933):896, 1994.

[5] Nikola Burić and Dragana Todorović. Dynamics of fitzhugh-nagumo excitable systems with delayed coupling. *Physical Review E*, 67(6):066222, 2003.

[6] George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314, 1989.

[7] 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.

[8] Richard FitzHugh. Impulses and physiological states in theoretical models of nerve membrane. *Biophysical journal*, 1(6):445–466, 1961.

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

[10] 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.

[11] 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.

[12] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

[13] Nirmalendu Hui. Cooperative behaviour of fitzhugh-nagumo system under direct-indirect coupling. *Journal of Scientific Research*, 66(1), 2022.

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

[15] Jinichi Nagumo, Suguru Arimoto, and Shuji Yoshizawa. An active pulse transmission line simulating nerve axon. *Proceedings of the IRE*, 50(10):2061–2070, 1962.

[16] Iku Nemoto, Shinji Miyazaki, Masao Saito, and Toshio Utsunomiya. Behavior of solutions of the hodgkin-huxley equations and its relation to properties of mechanoreceptors. *Biophysical Journal*, 15(5):469–479, 1975.

[17] 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.

[18] Ri-Qi Su, Ying-Cheng Lai, and Xiao Wang. Identifying chaotic fitzhugh–nagumo neurons using compressive sensing. *Entropy*, 16(7):3889–3902, 2014.

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

[20] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[21] Mike Wynne-Jones. Constructive algorithms and pruning: Improving the multi layer perceptron. In *Proceedings of the 13th IMACS World Congress on Computation and Applied Mathematics*, volume 2, pages 747–750, 1991.

[22] Binbin Xu, Stéphane Binczak, Sabir Jacquir, Oriol Pont, and Hussein Yahia. Parameters analysis of fitzhugh-nagumo

model for a reliable simulation. In *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 4334–4337. IEEE, 2014.