KADIR HAS UNIVERSITY
SCHOOL OF GRADUATE STUDIES
PROGRAM OF COMPUTATIONAL SCIENCES AND ENGINEERING

NETWORK RECONSTRUCTION FROM DATA

İREM TOPAL KEMENT

DOCTOR OF PHILOSOPHY THESIS

İSTANBUL, MAY, 2023

NETWORK RECONSTRUCTION FROM DATA

İREM TOPAL KEMENT
ADVISOR: ASST. PROF. DENİZ EROĞLU

A thesis submitted to
the School of Graduate Studies of Kadir Has University
in partial fulfilment of the requirements for the degree of
Doctor of Philosophy in
Computational Sciences and Engineering

İstanbul, May, 2023

APPROVAL

This thesis titled NETWORK RECONSTRUCTION FROM DATA submitted by İREM TOPAL KEMENT, in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Computational Sciences and Engineering is approved by

Asst. Prof. Deniz Eroğlu (Advisor) .....
Kadir Has University

Prof. A. Nihat Berker .....
Kadir Has University

Assoc. Prof. Alkan Kabakçıoğlu .....
Koç Univerisity

Prof. Uğur Tırnaklı .....
İzmir University of Economics

Assoc. Prof. Özkan Karabacak .....
Kadir Has University

I confirm that the signatures above belong to the aforementioned faculty members.

.....
Prof. Dr. Mehmet Timur Aydemir
Dean of School of Graduate Studies
Date of Approval: 11.05.2023

DECLARATION ON RESEARCH ETHICS AND PUBLISHING METHODS

I, İREM TOPAL KEMENT; hereby declare

- that this Ph.D. Thesis that I have submitted is entirely my own work and I have cited and referenced all material and results that are not my own in accordance with the rules;
- that this Ph.D. Thesis does not contain any material from any research submitted or accepted to obtain a degree or diploma at another educational institution;
- and that I commit and undertake to follow the "Kadir Has University Academic Codes and Conduct" prepared in accordance with the "Higher Education Council Codes of Conduct".

In addition, I acknowledge that any claim of irregularity that may arise in relation to this work will result in a disciplinary action in accordance with university legislation.

İREM TOPAL KEMENT

.....

11.05.2023

To the Academics for Peace

ACKNOWLEDGEMENT

My greatest thanks are to my advisor Asst. Prof. Deniz Eroğlu, for his guidance, assistance, and encouragement. He has always provided the best environment for me to do my best, and I am grateful for his understanding.

I am thankful to the thesis committee, Prof. A. Nihat Berker and Assoc. Prof. Alkan Kabakçıoğlu, for guiding me with constructive comments which improved the quality of this work.

I would like to thank Dr. Édgar Roldán and Dr. Jean Barbier for including me in their projects during the most difficult pandemic days. I will always gratefully remember the impact of the three months-visit to ICTP, Quantitative Life Sciences Section. I would also like to thank Dr. Roman Belousov, who made an invaluable contribution to that project with wise comments.

Being one of the first members of Nodds Lab has been a pleasure. I want to thank all current and former group members. Special thanks to my first office neighbor Emre Kaya, it was hard to get started but got easier thanks to him.

I owe a debt of gratitude to Mustafa Can Uslu, Azmi Can Özgen, Onur Kaplan and Taylan Özbay for the days full of Python, Linux and solidarity.

Many things would have been incomplete without the support of my dear friends: Neslişah Öncü, Hande Tınas Tomruk, Ferhan Nal Belgen, Hüseyin Belgen, Ayşe Nur Çelik, Barış Burnak, Sezen Yeşilaltay Saral, İ. Ceyhun Andaç, Serkan Türkoğlu, Eray Temur, Taygun Güngör, Arzu Balta, A. Ersin Öztürk, C. Yağmur Boynukara, Neslişah Bozkaya, Merve Elmas, Başak Kocadost, Kübra Çelikkaya, Yasemin Küçük. I hope we will meet again in Istanbul one day.

I am grateful to Dr. Elif Yunt, Dr. Meltem Ünel, Dr. Annwesha Dutta and Dr. Sarah A. M. Loos. I learned a lot from the days we empowered each other against the patriarchal system in academia and got along sometimes without words.

I am grateful to Dr. Onur Pusuluk, my role model, for accompanying me during all critical transitions in my life.

I am deeply grateful to Assoc. Prof. İrem İnceoğlu, Prof. Marc Timme, Dr. Pedro Harunari and Edmilson Roque dos Santos for motivating me to continue by adding

small perturbations.

I am thankful to my parents, Şengül Topal and Kemal Topal, for their constant caring and support in love. To Şebnem Topal, for a lovely childhood. To Nur Topal for always cheering us up since the day she was born. And to my beloved husband Erdi Kement for making life marvellous.

My special thanks are to the following people or groups who make me who I am, inspire me profoundly and support my journey through academia directly or indirectly: Mekansız Fizikçiler, Fizik Haftası, the Boğaziçi resistance, Prof. Ayşe Erzan, Prof. Ali Yıldız, Dr. Duygu Balcan, Kutsal Motor, Ünsal Ünlü, Nilay Örnek, Ozan Gündoğdu, Beatles, Led Zeppelin, Tame Impala, Ahmet Kaya, Ursula K. Le Guin, Latife Tekin, Selahattin Demirtaş, Ahmed Arif, Nazım Hikmet.

Finally, throughout the years of this thesis, many thanks to everyone who has supported me in thousands of different ways, motivated me even for a moment, stood up for academic freedom and resisted goings-on without losing their joy.

ABSTRACT

Dynamical networks, including power grids, food webs, climate networks, and neuron networks, described by dynamical units oscillating on complex networks, are fundamental components of our everyday lives. The ability to regulate network dynamics is crucial for predicting, thus, controlling these systems' behavior to acquire the desired functionality. Neuron networks are an important class of dynamical networks for human wellness since the changes in the interaction can lead to undesired pathological situations. For instance, epileptic seizures are associated with emergent neural network synchronization when dynamic network parameters change. Consequently, it is vital to anticipate critical transitions to neuronal synchronization and invent predictive technologies to detect early warning signals to prevent potential tragedies. In the case of neuron network dynamics, consisting of intrinsic neuron function and the coupling scheme between neurons, the critical transitions to synchronization are not directly determinable. Therefore, the governing equation must be recovered from the observations of the nodes for forecasting the critical transitions due to parameter changes. Therefore, the governing equation must be recovered from the observations of the nodes for forecasting the critical transitions due to parameter changes. This PhD thesis develops a dynamical network reconstruction approach from time series observations by integrating mean-field approaches from dynamical systems theory with statistical learning tools. The proposed reconstruction approach assumes a neuroscientific setting and accessibility to all nodes' data while the local dynamics of the nodes, the coupling function between them and the interaction structure are unknown. Our methodology accurately identifies them using relatively short time series and is independent of the network size, which is vital since it is generally impossible to have long real-world observations, and real networks are large. Finally, the reconstructed model allows us to predict the emergent collective behavior of dynamical networks considering parameter change, which

is crucial to avoid undesired behaviors for real-world applications such as epilepsy seizures.

Keywords: network dynamics, data, reconstruction, neuron networks, chaos, synchronization, collective behavior

ÖZET

Güç şebekeleri, ekosistem, iklim, nöron ağları ve bir hastalığın küresel ölçekte yayılması gibi hayatımızın temel bileşenlerinin bir ortak noktası vardır: karmaşık ağlar üzerinde etkileşen dinamik birimler olarak modellenebilmeleri. Pek çok örnekte, karmaşık sistemlerden elde edilen veriler doğal bir ağ yapısını temsil eder veya sistem özünde ağ yapısında olmasa bile bir ağ gibi modellenebilir. Ağ dinamiğini bilmek, bu karmaşık sistemlerden istenen işlevselliği elde etmek, dolayısıyla gelecekteki durumunu tahmin etmek ve kontrol etmek için çok önemlidir. Örneğin beynimizdeki nöron ağlarının etkileşimindeki normal olmayan değişiklikler patolojik durumlara yol açabileceğinden, bu ağlar insan sağlığı için önemli bir dinamik ağ sınıfını oluştururlar. Epilepsi krizleri nöron ağlarının etkileşimlerinin değişmesi ile beliren ağ senkronizasyonu ile ilişkilidir. Bu tip istenmeyen nöronal senkronizasyona kritik geçişleri önceden tahmin etmek ve erken uyarı sinyallerini tespit edecek teknolojileri icat etmek hayati önem taşır. Nöronların iç dinamikleri ve aralarındaki bağlantı şemasından oluşan nöron ağlarında, senkronizasyona kritik geçiş doğrudan belirlenemez. Bu nedenle amaç, parametre değişikliklerinden kaynaklanan kritik geçişleri tahmin etmek için ağ dinamiğinin denklemini her bir düğümden elde edilen ölçüm verisinden öğrenmektir. Bu doktora çalışması, dinamik sistemler teorisinden ortalama alan yaklaşımlarını istatistiksel öğrenme araçlarıyla birleştirerek zaman serisi gözlemlerinden dinamik bir ağı yeniden yapılandırma yaklaşımı sunar. Önerilen veri güdümlü yeniden yapılandırma yaklaşımı iki temel varsayımda bulunur: sinirbilimsel bir model ve tüm düğümlerin verisine tam erişim. Buna karşılık, düğümlerin iç dinamikleri, aralarındaki bağlantı yapısı ve etkileşim şekli bilinmez. Sinirbilimsel koşullar, nöronların iç dinamiğinin kaotik davranış göstermesi, zayıf bir etkileşimde olmaları ve ölçekten bağımsız bir ağ ile temsil edilmeleri olarak sıralanır. Metodolojimiz tüm bilinmeyenleri nispeten kısa zaman serileri kullanarak doğru bir şekilde öğrenir ve ağ boyutundan bağımsızdır. Kısa süreli ölçüm ve büyük ağlarda başarı

gerçek dünya örneklerine yaklaşabilmemiz için önemli iki kısıt olarak ele alınmıştır. Sonuç olarak, veriden öğrenilmiş ağ modeli tüm parametreleri kontrol edebilmemize ve karmaşık ağın kolektif davranışını tahmin edebilmemize izin verir.

Anahtar Sözcükler: ağ dinamiği, veri, yeniden yapılandırma, nöral ağlar, kaos, senkronizasyon, kolektif davranış

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

r	Control parameter for logistic map
n	System size
m	Dimension of the local dynamics
T	Length of the time series
w_{ij}	Elements of the adjacency matrix
L_{ij}	Elements of the Laplacian matrix
k_i	Incoming degree of node i
d_{ij}	Euclidean distance between predicted models of node i and j
K	Sparsity measure of the network
η_i	Noise term for node i
α	Multiplier for the coupling function
β	Control parameter of Rulkov map
ν	Control parameter of Rulkov map
σ	Control parameter of Rulkov map
μ	A physical measure of the isolated dynamics
κ_i	Negligible fluctuation on node i
ψ	Candidate function in the basis library
ξ	Coefficient vector
ϵ	Tolerance value for error function
γ	Coupling control parameter

LIST OF ACRONYMS AND ABBREVIATIONS

AA	Arterial aneurysms
AVM	Arteriovenous malformations
FN	False negatives
FNR	False negative rate
FP	False positives
FPR	False positive rate
LASSO	The least absolute shrinkage and selection operator
mse	The mean squared error
N	Negatives
P	Positives
RSS	Residual sum of squares
SINDy	Sparse identification of nonlinear dynamics
STLS	Sequential thresholded least squares

1. INTRODUCTION

We are surrounded by complex networks composed of many interacting units that form an intricate interaction structure by individually interacting with one or many others. Examples of complex systems are power grids consisting of generators and loads, a climate system of interacting spatially-diverse weather patterns and the brain and its neuronal networks. On the other hand, recent technological advances have made collecting large amounts of data from such complex systems possible. However, more than data availability is needed to understand a complex system's behavior better. A major scientific challenge is to develop methods for understanding the mechanism behind the collective dynamics of a complex network so that we could be able to predict critical transitions. Throughout the thesis, we seek to reveal the network structure among units and the rules that govern the underlying dynamics from spatiotemporal data. Reconstructing network dynamics from data is crucial for predicting the changes in the dynamics of complex systems such as neuron networks. For instance, synchronization is the most prominent collective behavior observed due to neuronal interactions. Some brain disorders like Parkinson's, Alzheimer's and epilepsy are associated with abnormal neural network synchronization (Rakshit *et al.*, 2018). Modeling and analyzing the emergent behavior of a complex network became the main direction of the research on network dynamics. Then, the reverse questions were addressed by the researchers from the natural sciences (Timme and Casadiego, 2014). However, previous research has shown that the reconstruction is possible under strong constraints such as the need for lengthy data or small system size. This study presents a recovery scheme blending theoretical model reduction and sparse recovery to identify the governing equations and the interactions of weakly coupled chaotic maps on complex networks, easing unrealistic constraints for real-world applications. Learning dynamics and connectivity lead to detecting critical transitions for parameter changes. We apply our technique to

realistic neuronal systems with and without noise on a real mouse neocortex and artificial networks. Before briefly reviewing background material of the thesis study, some relevant history will be given in this section.

Dynamics, as a branch of physics, began in the mid-1600s with the invention of differential equations by Newton. He discovered laws of motion and universal gravitation. By combining them, specifically, by solving the two-body problem, he explained Kepler's laws of planetary motion. The two-body problem was calculating the Earth's motion around the sun. Extension of Newton's analytical methods to solve the three-body problem (sun, earth and moon) had been tried by subsequent generations of mathematicians and physicists. Finally, it was seen that solving this problem by obtaining explicit formulas for the motions of three bodies is impossible. Poincare made a breakthrough with his work in the late 1800s by approaching the problem in a qualitative rather than a quantitative way. He developed a geometric approach by asking questions from a qualitative perspective, like the solar system's stability instead of the planets' exact positions. Although Poincare was the first person to realize the possibility of *chaos*, most of the effort spent nonlinear oscillations in the first half of that century. Works of nonlinear oscillators helped both the development of new technologies such as laser and the invention of new mathematical techniques (Strogatz, 2000).

The history of dynamics has a milestone in the 1950s: the invention of the high-speed computer. Lorenz discovered chaotic motion in 1963 (Lorenz, 1963) utilizing the developments of nonlinear systems. He was working on a simplified atmosphere model and found that the solutions to his model never settled down to equilibrium or to a periodic state. Also, he realized the *sensitive dependence on initial conditions* and the amplification of tiny measurement errors in the state of the atmosphere. Moreover, he found the structure in the chaos and is called *fractal* today (Strogatz, 2000).

Modern *dynamical systems* is a well-developed branch of mathematics and physics.

It is used to model real-world phenomena for future state prediction, optimization, control and physical understanding. Today, data-driven approaches are leading the improvements in analysis and understanding of dynamical systems. There are abundant data while governing equations are still elusive for many fields such as climate, epidemiology and neuroscience. Parallel to the developments in data science, *data-driven dynamical systems* field is evolving rapidly (Brunton and Kutz, 2019).

This thesis study includes four chapters. *Introduction* chapter introduces the fundamental theories of dynamical systems based on discrete-time systems. Then, the basics of statistical learning tools are described to set the stage for the model discovery of dynamical systems in a data-driven way. The last section is devoted to synchronization phenomena from coupled dynamical systems to complex networks by giving the main elements of the network theory. *Theory and Modeling* chapter of the thesis provides the main neuroscientific setting by introducing the dynamical system to mimic neuron behavior, used networks, reduction theorem from dynamical systems and the proposed methodology for network reconstruction. Results of the numerical experiments and example dynamical systems are given in *Results* chapter. The last chapter presents the conclusions and main discussions on open questions in the field.

1.1 Dynamical Systems

In this chapter we will investigate the fundamentals of discrete dynamical systems.

1.1.1 Discrete dynamical systems

We consider dynamical systems of the form

$$x(t+1) = f(x(t)) \tag{1.1}$$

where $x \in \mathbb{R}$ and for some smooth function $f\colon \mathbb{R} \to \mathbb{R}$. These systems are also known as iterated maps or simply *maps*. The initial value of x at initial time ($t = 0$)

is given for the iteration. As an example, consider the logistic map

$$x(t+1) = rx(t)(1 - x(t)) \tag{1.2}$$

which is a discrete-time analog of the logistic equation for population growth and $r \geq 0$ is the intrinsic growth rate (Strogatz, 2000). As shown in Fig. 1.1, Eq. (1.2) gives a parabola with a maximum of $r/4$ at $x = 1/2$.

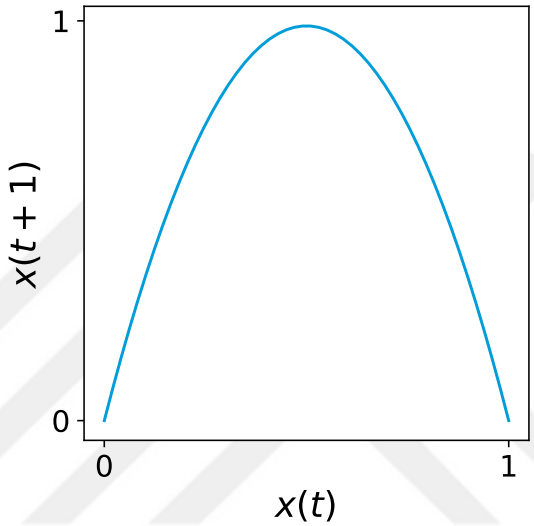

Figure 1.1 Logistic map for $r = 3.95$. 50 iterates are plotted.

We focus on the properties of *fixed points* corresponding to a steady state observed in a realistic system. The stability of fixed points is essential since a real-world system is constantly exposed to small perturbations (Alligood *et al.*, 2000). $x = x^*$ is a fixed point if it satisfies $f(x^*) = x^*$. If the system reaches x^*, then it remains at x^* and does not change over time. The stability of x^* is determined by considering the dynamics close to that fixed point. A nearby orbit is chosen as $x = x^* + \eta$ where η represents a small distance from the fixed point. As time evolves growing or decaying of η will determine the stability or instability (Newman, 2010).

$$x^* + \eta(t+1) = x(t+1) = f(x^* + \eta(t)) = f(x^*) + f'(x^*)\eta(t) + \mathcal{O}(\eta(t)^2) \tag{1.3}$$

where $f'(x^*)$ is the derivative of f evaluated at x^* and $\mathcal{O}(\eta(t)^2)$ denotes quadratically small terms in η. Since $f(x^*) = x^*$, by neglecting $\mathcal{O}(\eta(t)^2)$ terms, Eq. (1.3) reduces

to

$$\eta(t+1) = f'(x^*)\eta(t) \tag{1.4}$$

Thus, we obtain the *linearized map* in Eq. (1.4) with eigenvalue or multiplier $\lambda = f'(x^*)$. By writing a few terms explicitly, the solution of this linear map can be found: $\eta(1) = \lambda\eta(0)$, $\eta(2) = \lambda\eta(1) = \lambda^2\eta(0)$, and in general $\eta(t) = \lambda^t\eta(0)$. Note that λ is just a simple number which can be found if x^* and f is provided. The distance η from the fixed point will decay as $t \to \infty$ if $|f'(x^*)| < 1$ and the fixed point is *linearly stable*. Conversely, if $|f'(x^*)| > 1$, then the fixed point is *unstable*. This analysis of the fixed points is *linear stability analysis*. One can generalize these results to systems with two or more variables. Consider a two-dimensional discrete time dynamical system as

$$\begin{aligned} x(t+1) &= f(x(t), y(t)) \\ y(t+1) &= g(x(t), y(t)) \end{aligned} \tag{1.5}$$

A fixed point at (x^*, y^*) satisfies $f(x^*, y^*) = x^*$ and $g(x^*, y^*) = y^*$. By neglecting the quadratic terms again, the time evolution of (η_x, η_y) which is close to the fixed point in two-dimensional space is written as

$$\eta_x(t+1) = f^{(x)}(x^*, y^*)\eta_x(t) + f^{(y)}(x^*, y^*)\eta_y(t) \tag{1.6}$$

$$\eta_y(t+1) = g^{(x)}(x^*, y^*)\eta_x(t) + g^{(y)}(x^*, y^*)\eta_y(t) \tag{1.7}$$

where $f^{(x)}$, $g^{(x)}$ and $f^{(y)}$, $g^{(y)}$ are the derivatives of f and g with respect to x and y. By combining Eqs. (1.6) and (1.7) we write them in the matrix form

$$\boldsymbol{\eta}(t+1) = \boldsymbol{J}\boldsymbol{\eta}(t) \tag{1.8}$$

where $\boldsymbol{\eta}$ is the two-dimensional vector (η_x, η_y) and $\boldsymbol{J}$ is the Jacobian matrix

$$\boldsymbol{J} = \begin{pmatrix} \frac{\partial f}{\partial x} & \frac{\partial f}{\partial y} \\ \frac{\partial g}{\partial x} & \frac{\partial g}{\partial y} \end{pmatrix},$$

where the partial derivatives are all calculated at the fixed point (x^*, y^*) (Newman, 2010). The linearization is given by the derivative at the fixed point in one dimensional case, now it is provided by a matrix for higher-dimensional case (Alligood *et al.*, 2000). We highly desire to linearize the nonlinear dynamical system near a fixed point, since the system turns into eigenvector coordinates where the dynamics become decoupled. The dynamics are characterized by the eigenvalues and the eigenvectors of the matrix $\boldsymbol{J}$ and there are many techniques for the analysis, prediction or control of such systems (Brunton and Kutz, 2019).

1.1.2 Chaos

Chaos is identified by three ingredients:

1. "Aperiodic long-term behavior" meaning as $t \to \infty$ the trajectories do not settle down to fixed points or periodic orbits.
2. "Deterministic" meaning that the system shows irregular behavior due to its nonlinearity, it has no random or noisy inputs.
3. "Sensitive dependence on initial conditions" meaning that nearby (starting very close) trajectories will separate exponentially fast (Strogatz, 2000).

Robert May worked on a version of the logistic difference equation in the 1970s. He discovered that using different values for the nonlinear parameter could change the system's behavior completely (Gleick, 1987). The results of his calculation using a bifurcation diagram, as shown in Fig. 1.2(a), state that simple mathematical models also exhibit complicated behavior (May, 1976).

Fig. 1.2(a) shows the bifurcation diagram of logistic map for the range $0 \leq r \leq 4$. Then, the control parameter r is fixed at some certain values and the results are plotted as time series. For $1 < r < 3$, system reaches a nonzero steady state and it can be seen in Fig. 1.2(b) for $r = 2.8$ as time series. For $r = 3.3$ in Fig. 1.2(c), we see *period-2 cycle*. In Fig. 1.2(d), for $r = 3.5$ we see that the previous cycle has doubled its period to *period-4*. As seen in the bifurcation diagram, for many values

of r, system does not settle down to a fixed point or a periodic orbit. For these values, the motion is aperiodic for long-term as seen in Fig. 1.2(d) and this is the discrete-time version of chaos (Strogatz, 2000).

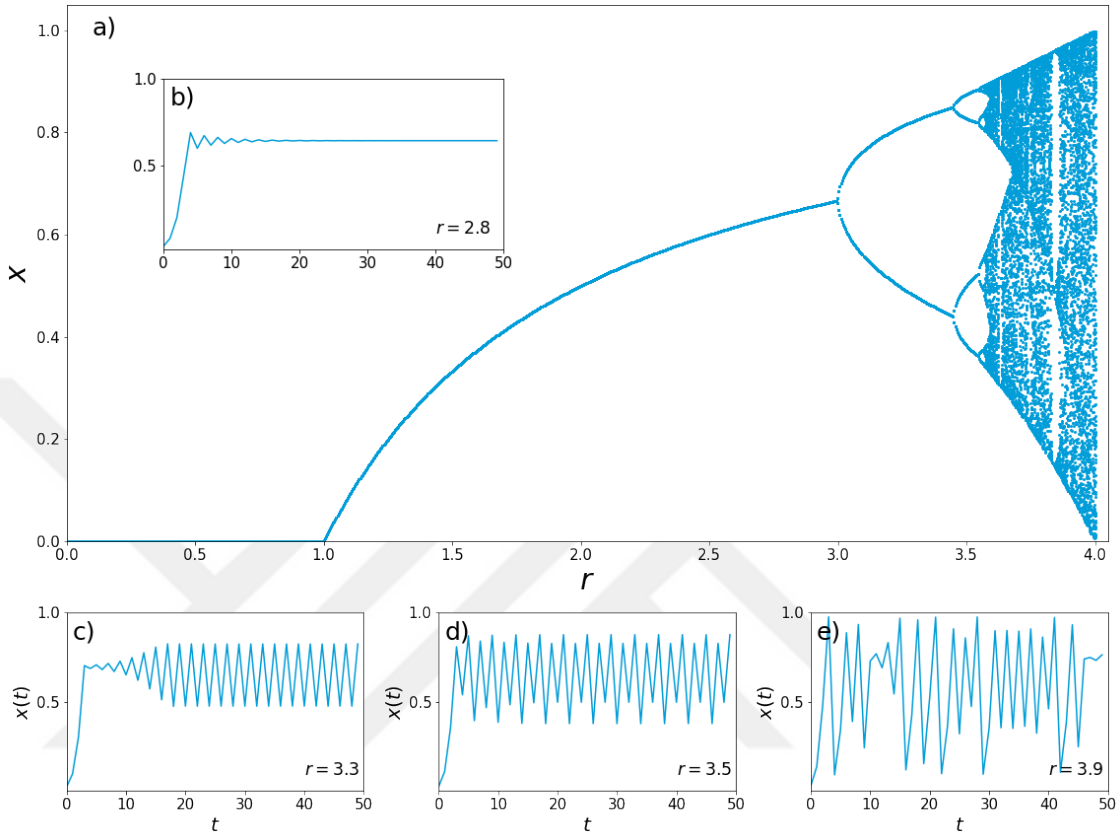

Figure 1.2 a) Bifurcation diagram of logistic map. Time series results for b) $r = 2.8$, c) $r = 3.3$, d)$r = 3.5$ and e)$r = 3.9$.

The logistic map analysis will be continued by showing the sensitive dependence on the initial conditions. Suppose we have a point $x(t)$ and a neighboring point is chosen as $x(t)+\delta(t)$ where δ is a small separation vector of initial length $\|\delta(0)\| = 10^{-15}$. We will check two neighboring orbits' separation from growing of δ after t iterations. If $\|\delta(t)\| = \|\delta(0)\|e^{t\lambda}$), then λ is called as *Lyapunov exponent* and a positive exponent causes separation of the close orbits exponentially fast.

To compute λ, a formula is derived by taking logarithms and noting that $\delta(t) =$

$f^t(x(0)+\delta(0)) - f^t(x(0))$

$$\begin{aligned}\lambda &\approx \frac{1}{t}\ln\left|\frac{\delta(t)}{\delta(0)}\right| \\ &= \frac{1}{t}\ln\left|\frac{f^t(x(0)+\delta(0)) - f^t(x(0))}{\delta(0)}\right| \\ &= \frac{1}{t}\ln|(f^t)'(x(0))| \end{aligned} \tag{1.9}$$

in the limit $\delta(0) \to 0$. By the chain rule, the logarithm is expanded as:

$$(f^t)'(x(0)) = \prod_{i=0}^{t-1} f'(x_i) \tag{1.10}$$

Then,

$$\begin{aligned}\lambda &\approx \frac{1}{t}\ln\left|\prod_{i=0}^{t-1} f'(x_i)\right| \\ &= \frac{1}{t}\sum_{i=0}^{t-1}\ln|f'(x_i)|. \end{aligned} \tag{1.11}$$

We present Lyapunov exponent results for logistic map computed numerically with Eq. (1.11) in Fig. 1.3. One can compare this results with the bifurcation diagram in Fig. 1.2(a). Lyapunov exponent λ is negative for $r < 3.57$. Chaos can be seen near $r \approx 3.57$, where λ becomes positive for the first time. Even tough Lyapunov exponent increases generally after that value, there exist some dips caused by the windows of periodic motion (Strogatz, 2000).

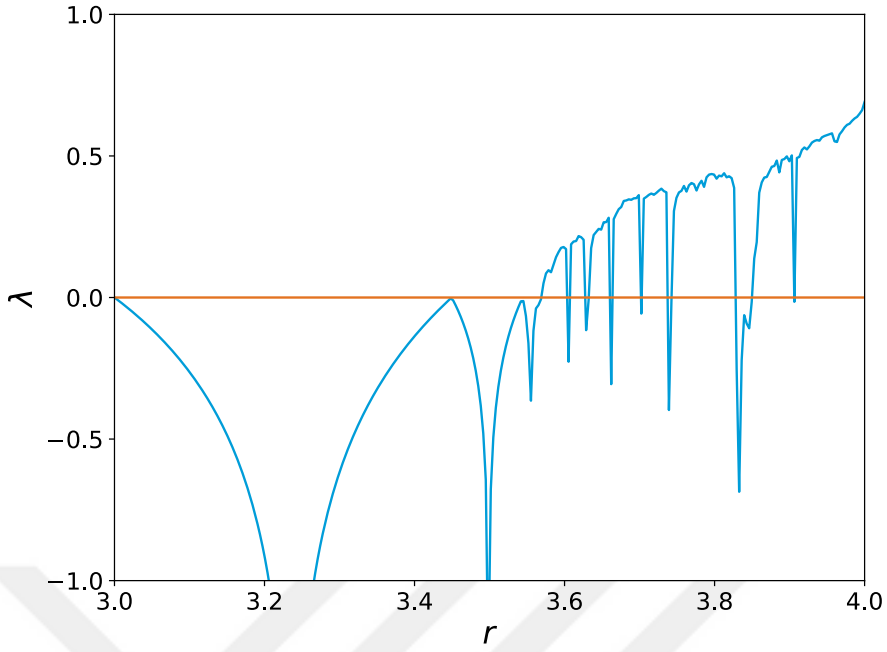

Figure 1.3 Lyapunov exponents of the logistic map versus bifurcation parameter r. The exponent increases and hits 0 at period-2 and becomes positive in the chaotic region.

1.2 Statistical Learning

Statistical learning refers to a set of tools for understanding complex datasets. These tools are classified as supervised and unsupervised mainly. Supervised statistical learning involves modeling data to predict an output based on one or more inputs. Unsupervised statistical learning aims to discover relationships and structure from data when there is no supervising output (James *et al.*, 2014).

The term *statistical learning* is fairly new, but many concepts like classic curve fitting under the discipline are old (James *et al.*, 2014). Least-squares is used to fit astronomical data by Legendre in 1805 (Legendre, 1805) and its theory as an optimization problem is fully developed by Gauss in 1821 (Gauss, 1823). Classic curve fitting with the method of least squares as one of the most fundamental tools in data science is the earliest form of *regression*. Regression techniques mainly shape data science and some of these techniques which are mostly used in the thesis will be given throughout this section.

1.2.1 Linear regression

Linear regression is a widely used statistical learning tool to predict a quantitative response. Y is predicted by the linear model

$$\hat{Y} = \hat{\beta}_0 + \sum_{j=1}^{p} X_j \hat{\beta}_j \tag{1.12}$$

where $\boldsymbol{X}^T = (X_1, X_2, \cdots, X_p)$ is the given vector of inputs. Two unknown parameters β_0 and β_j are the intercept and slope terms, respectively. It is common to have the constant variable 1 in X, then including β_0 in β, we rewrite the linear model Eq. (1.12) in vector form as an inner product:

$$\hat{Y} = \boldsymbol{X}^T \hat{\beta}. \tag{1.13}$$

where $\boldsymbol{X}^T$ denotes vector or matrix transpose and $\hat{\beta}$ is known as the model coefficients or parameters (James *et al.*, 2014; Hastie *et al.*, 2009). Our aim is to find the linear model fits the training data well. We find the coefficients β to minimize the residual sum of squares

$$RSS(\beta) = \sum_{i=1}^{N} (y_i - x_i^T \beta)^2 \tag{1.14}$$

and it is called classic *least squares*. $RSS(\beta)$ is a quadratic function of the model parameters which means it has minimum, but may not be unique. We rewrite Eq. (1.14) as

$$RSS(\beta) = (\boldsymbol{y} - \boldsymbol{X}\beta)^T (\boldsymbol{y} - \boldsymbol{X}\beta) \tag{1.15}$$

Least squares error function is differentiated w.r.t β and we obtain

$$\boldsymbol{X}^T (\boldsymbol{y} - \boldsymbol{X}\beta) = 0 \tag{1.16}$$

The unique solution is given by

$$\hat{\beta} = (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{y} \tag{1.17}$$

where $\boldsymbol{X}^T \boldsymbol{X}$ is nonsingular (Hastie *et al.*, 2009).

1.2.2 Over- and under-determined systems

Typically, the linear system we want to solve is massively over- or under-determined in data science problems. Over-determined systems have more constraints (equations) than unknown variables (Fig. 1.4(a)). Thus, the linear system is approximately solved to minimize a given error function. Under-determined systems have more unknowns than constraints (Fig. 1.4(b)), so there exist an infinite number of solutions, and to obtain a unique solution, some conditions must be applied. To solve $Ax = b$ for over- and under-determined systems, ℓ_1 or ℓ_2 norms can be used, and the choice of the norm has a profound impact on the optimal solution achieved (Brunton and Kutz, 2019).

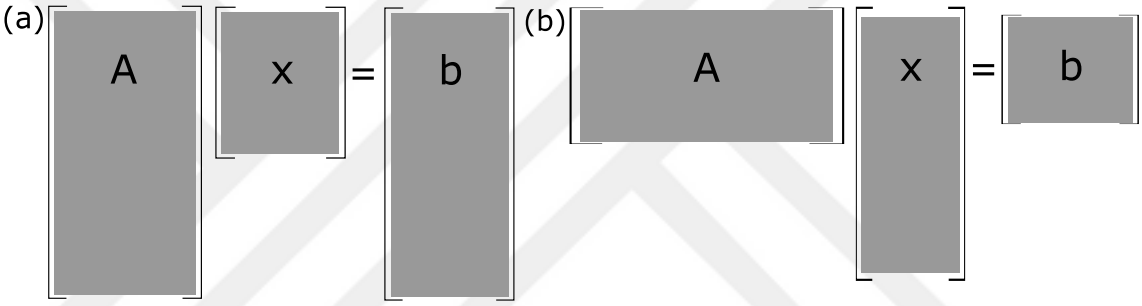

Figure 1.4 Regression framework for a) over-determined system and b) under-determined system.

1.2.3 Shrinkage methods

The aim is to use a technique that constrains the coefficient estimates, or shrinks them towards zero to obtain an interpretable model. Shrinking the regression coefficients decrease their variance (James *et al.*, 2014). The two best-known shrinkage methods are presented as *Ridge regression* and *the Lasso*.

Ridge regression. A penalty term is added to the residual sum of least squares

$$\sum_{i=1}^{N}(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \tag{1.18}$$

where λ is penalty term which controls the amount of shrinkage. Selecting a good value for λ is vital, a large penalty term will decrease the magnitudes of the coeffi-

cients (Hastie *et al.*, 2009).

The least absolute shrinkage and selection operator (Lasso). The penalty term in Eq. (1.18) will shrink the estimated coefficients towards zero but any of them will not be set exactly to zero in Ridge regression. When the number of variables is large, it is challenging to interpret a model since Ridge regression includes all predictors in the final model. The lasso coefficients minimize

$$\sum_{i=1}^{N}(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \tag{1.19}$$

which uses ℓ_1-norm to penalize the coefficients (James *et al.*, 2014). The ℓ_1-norm of a coefficient vector is $\|\beta\|_1 = \sum |\beta_j|$. By forcing some of the coefficients to be equal to zero, Lasso performs variable selection and generates interpretable models.

1.2.4 Sparse identification of nonlinear dynamics (SINDy)

Discovering the nonlinear nature of a model from data is a challenging problem since there are many possible model structures. The sparse identification of nonlinear dynamics (SINDy) states the fact that many dynamical systems

$$\frac{d}{dt}\boldsymbol{x} = \boldsymbol{f}(\boldsymbol{x}) \tag{1.20}$$

have only a few terms in the space of candidate right-hand side functions. Then, the model for the dynamical system is generalized by the function $\boldsymbol{f}$:

$$\boldsymbol{f}(\boldsymbol{x}) \approx \sum_{k=1}^{p} \psi_k(\boldsymbol{x})\xi^k \tag{1.21}$$

where $\boldsymbol{\Psi}(\boldsymbol{x}) = [\psi_1(\boldsymbol{x}), \psi_2(\boldsymbol{x}), ..., \psi_p(\boldsymbol{x})]$ is the library of basis functions. Only a few active terms characterize data that comes from natural systems on a well-chosen basis; most of the coefficients ξ^k are zero (Brunton and Kutz, 2019). To find interpretable models sparsity concept is beneficial. This idea was used in a compressed sensing framework for the first time as the sparsity requirement is satisfied (Wang, Yang, Lai, Kovanis and Grebogi, 2011). If we have prior knowledge to decide the basis, sparse regression is instrumental in avoiding overfitting and noise (Brunton

et al., 2016). For discrete time dynamical systems, SINDy algorithm is formulated as

$$\boldsymbol{x}_{t+1} = \boldsymbol{f}(\boldsymbol{x}_t) \tag{1.22}$$

and the data matrix of derivatives is avoided entirely (Brunton and Kutz, 2019).

LASSO (Tibshirani, 1996) as a sparse regression method is used to obtain *sparse* models. Sequentially-thresholded least-squares (STLS) is another method implemented in the Python package of SINDy (de Silva *et al.*, 2020) and used in SINDy framework. In STLS, Ridge regression is used and λ is fixed as 0.05 (Eq, (1.18)). Sparsity is obtained by masking out elements of $\hat{\boldsymbol{\beta}}$ that are below a given threshold and an interpretable dynamical model is obtained. This threshold is called as *sparsity parameter* and it is the single hyper-parameter of STLS.

1.2.5 Logistic regression

In many cases, the response variable is *qualitative* and they are referred as *categorical*. To predict the qualitative response of an observation is *classifying* that observation, since a class is assigned to the observation. In the case of a qualitative response, linear regression is not appropriate since we do not want to model this response. *Logistic regression* models the probability that the response belong to a particular class. Logistic function is used

$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}} \tag{1.23}$$

Maximum likelihood method is used to estimate the model coefficients (James *et al.*, 2014).

1.3 Coupled Dynamical Systems

Consider two identical discrete maps with $x(t+1) = f(x(t))$ and $y(t+1) = f(y(t))$. Suppose, in the independent case, the dynamics of each variable is chaotic. Now, we introduce an interaction between the two maps by adding a term on the right-hand

side of the equation which contains both variables as (Pikovsky *et al.*, 2001)

$$\begin{aligned} x(t+1) &= f(x(t)) + \alpha(f(y(t)) - f(x(t))) \\ y(t+1) &= f(y(t)) + \alpha(f(x(t)) - f(y(t))) \end{aligned} \tag{1.24}$$

where α is the coupling strength. By rewriting Eq. (1.24) in matrix form, we obtain

$$\begin{bmatrix} x(t+1) \\ y(t+1) \end{bmatrix} = \begin{bmatrix} (1-\alpha) & \alpha \\ \alpha & (1-\alpha) \end{bmatrix} \begin{bmatrix} f(x(t)) \\ f(y(t)) \end{bmatrix}. \tag{1.25}$$

There is no unique way to add an interaction between two maps, for instance, we could couple them as (Lloyd, 1995),

$$\begin{aligned} x(t+1) &= f(x(t)) + \alpha(y(t) - x(t)) \\ y(t+1) &= f(y(t)) + \alpha(x(t) - y(t)). \end{aligned} \tag{1.26}$$

For the varying positive coupling parameter α in Eq. (1.25), one can describe the observations qualitatively. In $\alpha = 0$ case, the system becomes two isolated discrete maps. Two maps can be coupled weakly or relatively strong depending on the value α. If the coupling parameter is considered as a bifurcation parameter and assume we increase it gradually from zero, we observe a complex bifurcation structure. After a critical α value, the system *synchronizes* which we will discuss later (Pikovsky *et al.*, 2001).

1.4 Synchronization

Synchronization is related to the study of dynamical stability (Newman, 2010). Any oscillator can be used to model many systems of scientific interest. Radio communication, fireflies flashing, birds flapping their wings, clapping of a large audience, the heart itself, pathologically shaking limbs due to Parkinson's disease and firing of brain cells - a common feature of these systems is they have rhythms. They interact with their surroundings weakly or strongly and adjust their rhythm with the rhythms of others. Rhythm adjustment due to an interaction is the source of synchronization (Pikovsky *et al.*, 2001).

We introduced Logistic map as the simplest chaotic system in Section 1.1.1. In this section, two-coupled logistic map model will be given to describe the synchronization transition phenomenologically. We rewrite the Eq. (1.24) for logistic map as

$$\begin{aligned} x(t+1) &= x(t)r(1-x(t)) + \alpha[y(t)r(1-y(t)) - x(t)r(1-x(t))] \\ y(t+1) &= y(t)r(1-y(t)) + \alpha[x(t)r(1-x(t)) - y(t)r(1-y(t))] \end{aligned} \tag{1.27}$$

where $r = 3.6$. In the context of this thesis, we say a system of two coupled oscillators x and y synchronizes if the following two properties hold (Eroglu *et al.*, 2017):

- If $x(0) = y(0)$, then $x(t) = y(t)$ for all t.
- If $x(0)$ is close to $y(0)$, then $\lim_{t\to\infty}[x(t) - y(t)] = 0$.

Initial conditions are $x(0) = 0.54$ and $y(0) = 0.55$. Two coupled maps are iterated for 10000 steps and after transient, final 100 steps are taken for the analysis. Fig. 1.5 shows the synchronization error between two time series which is given by $\frac{1}{T}\|x(t) - y(t)\|$ with respect to a series of α values. For this particular system, we see synchronization in the range $0.38 \leq \alpha \leq 0.81$ as the difference between two time series disappears. Inset in Fig.1.5 shows $\alpha = 0.4$ case as time series. We see transition to synchronization after a very short time.

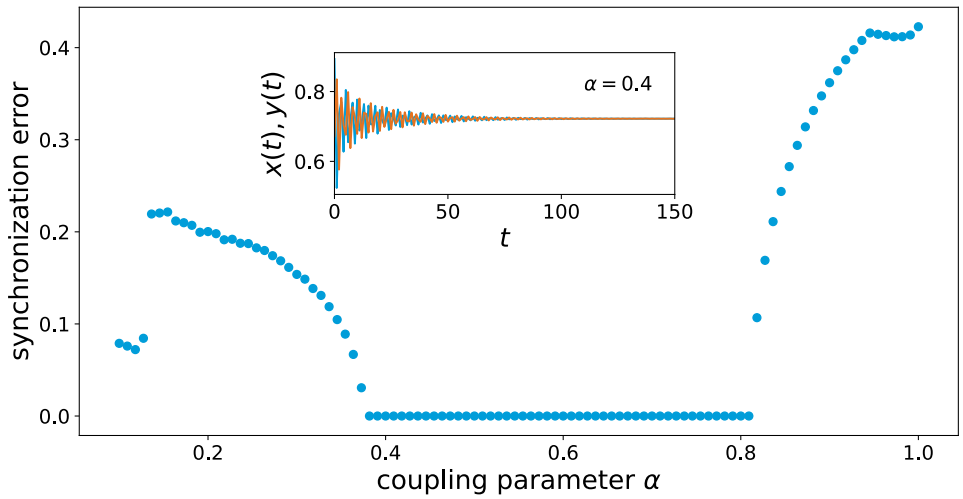

Figure 1.5 Two coupled logistic maps for $r = 3.6$. As synchronization error between two time series tends to zero, critical coupling parameter region for synchronization is determined. Inset shows the transition for $\alpha = 0.4$

One can model the pathologically synchronized firing neurons during an epileptic seizure with *complex networks* of coupled oscillators so that the coupling could cause the synchronization.

1.5 Complex Networks

The roots of network science go back to 1736 for one of the oldest proofs in graph theory. The problem was to find a route that goes through all seven bridges of Königsberg while never crossing the same bridge twice. Leonard Euler proved that such a path does not exist and he was the first who solved a mathematical problem using a graph. To understand a complex system, one need to know the nature of the interacting components first, in other words, the connectivity structure (Barabási and Pósfai, 2016). The basics of the network science used throughout this thesis will be given in this section.

1.5.1 Basics

A *network* is a collection of nodes joined by links (Newman, 2010). The link (i, j) which joins the nodes i and j represent the relation between these nodes. This relation can be social, physical, conceptual, chemical, biological or any other, therefore networks allow studying interrelations in a wide range of systems by providing a general theoretical framework (Menczer *et al.*, 2020).

The adjacency matrix $\boldsymbol{A}$ is a way to represent a network mathematically. $\boldsymbol{A}$ is a matrix with elements

$$A_{ij} = \begin{cases} 1 & \text{if there is a link between nodes } i \text{ and } j \\ 0 & \text{otherwise} \end{cases} \tag{1.28}$$

In some cases, links have strength, weight or value, and such *weighted networks'* elements of adjacency matrices are equal to the weights of the corresponding links. The weighted nature of the networks is given by different widths of lines in the graphical representation. In *directed networks*, each link has a direction and is shown

by an arrow pointing from one node to another. The number of links connected to a node is the *degree* of that node, and it is denoted by k_i for node i. For an undirected network of n nodes, the degree of node i is given by (Newman, 2010; Menczer *et al.*, 2020)

$$k_i = \sum_{j=1}^{n} A_{ij}. \tag{1.29}$$

The graph Laplacian is another matrix to represent a network mathematically and is related to the adjacency matrix. We will explain this matrix by considering diffusion processes on networks. Diffusion is a useful way of thinking about spreading of any quantity across a network simply. Suppose modeling a commodity moves along the links and there is an amount of ψ_i of it at node i. Then, the amount of commodity moving from j to i in a small interval of time is given by

$$C(\psi_j - \psi_i)dt \tag{1.30}$$

where C is diffusion constant. The rate of changing at ψ_i is given by

$$\frac{d\psi_i}{dt} = C\sum_j A_{ij}(\psi_j - \psi_i), \tag{1.31}$$

and with the adjacency matrix in Eq. (1.31) only neighboring nodes of node i are appeared in the sum. Then, we rewrite the equation by splitting the two terms

$$\begin{aligned} \frac{d\psi_i}{dt} &= C\sum_j A_{ij}\psi_j - C\psi_i\sum_j A_{ij} = C\sum_j A_{ij}\psi_j - C\psi_i k_i \\ &= C\sum_j (A_{ij} - \delta_{ij}k_i)\psi_j \end{aligned} \tag{1.32}$$

where δ_{ij} is the Kronecker delta (equal to 1 if $i = j$ and 0 otherwise). In matrix form

$$\frac{d\boldsymbol{\psi}}{dt} = C(\boldsymbol{A} - \boldsymbol{D})\boldsymbol{\psi} \tag{1.33}$$

where $\boldsymbol{\psi}$ vector has the elements ψ_i, $\boldsymbol{A}$ is the adjacency matrix and $\boldsymbol{D}$ is the diagonal matrix which has the node degrees along its diagonal such:

$$\boldsymbol{D} = \begin{pmatrix} k_1 & 0 & 0 & \dots \\ 0 & k_2 & 0 & \dots \\ 0 & 0 & k_3 & \dots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}$$

By defining $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{A}$, Eq. (1.32) is written as

$$\frac{d\boldsymbol{\psi}}{dt} + C\boldsymbol{L}\boldsymbol{\psi} = 0 \tag{1.34}$$

If the Laplacian operator ∇^2 is replaced by the matrix $\boldsymbol{L}$ in Eq. (1.34), it becomes ordinary diffusion equation, therefore $\boldsymbol{L}$ is called the graph Laplacian. The elements of the Laplacian matrix is written as

$$L_{ij} = \begin{cases} k_i & \text{if } i = j \\ -1 & \text{if } i \neq j \text{ and there is a link } (i,j) \\ 0 & \text{otherwise} \end{cases} \tag{1.35}$$

Therefore, we can write $L_{ij} = \delta_{ij}k_i - A_{ij}$ (Newman, 2010).

One of the most fundamental properties of networks is *degree distribution*. p_k is the fraction of the nodes with degree k in a network, and we plot k vs p_k to see the frequency of node degrees. Distribution of the form

$$p_k = Ck^{-\alpha} \tag{1.36}$$

where C is a constant and α is the exponent are called power laws and networks with power law degree distributions are called *scale free networks* (Newman, 2010).

If each node in a network is connected to other nodes, then the network has the maximum number of links and it has the maximum *density*. However, in most real-world networks, the actual number of links is typically much smaller than the maximum. This property is called *sparsity* and as a natural feature it helps to deal with network structure (Menczer *et al.*, 2020).

1.5.2 Network dynamics

The network dynamics of weakly coupled and identical n oscillators with interaction akin to diffusion is described by

$$\boldsymbol{x}_i(t+1) = \boldsymbol{f}(\boldsymbol{x}_i(t)) + \sum_{j=1}^{n} w_{ij}\boldsymbol{H}(\boldsymbol{x}_i(t), \boldsymbol{x}_j(t)) + \boldsymbol{\eta}_i(t) \tag{1.37}$$

where $\boldsymbol{x}_i \in \mathbb{R}^m$, $\boldsymbol{f}\colon \mathbb{R}^m \to \mathbb{R}^m$ represents the isolated dynamics of nodes and we assume it is chaotic (Barzel and Barabási, 2013). $\boldsymbol{H}$ is a diffusive coupling function ($\boldsymbol{H}(\boldsymbol{x}, \boldsymbol{x}) = \boldsymbol{H}(0) = 0$ and $\boldsymbol{H}(\boldsymbol{x}, \boldsymbol{y}) = -\boldsymbol{H}(\boldsymbol{y}, \boldsymbol{x})$). $\boldsymbol{W} = [w_{ij}] \in \mathbb{R}^{n\times n}$ is the adjacency matrix of weighted and directed network where $w_{ij} \geq 0$ is the interaction strength from node-j to node-i. The noise term, $\boldsymbol{\eta}_i(t)$, is uniformly distributed $\|\boldsymbol{\eta}_i(t)\| \leq \eta_0$ for all nodes where η_0 is noise intensity. This network dynamics, Eq. (1.37), is used to model numerous real-world applications including brain networks (Izhikevich, 2007), power grids (Dörfler *et al.*, 2013; Motter *et al.*, 2013), superconductors (Watanabe and Strogatz, 1994), and cardiac pacemaker cells (Winfree, 2001). Diffusive nature of the interaction allows represent the coupling in terms of the Laplacian matrix.

$$\begin{aligned}
\sum_{j=1}^{n} w_{ij}[\boldsymbol{H}(\boldsymbol{x}_j) - \boldsymbol{H}(\boldsymbol{x}_i)] &= \sum_{j=1}^{n} w_{ij}\boldsymbol{H}(\boldsymbol{x}_j) - \boldsymbol{H}(\boldsymbol{x}_i)\sum_{j=1}^{n} w_{ij} \\
&= \sum_{j=1}^{n} w_{ij}\boldsymbol{H}(\boldsymbol{x}_j) - k_i\boldsymbol{H}(\boldsymbol{x}_i) \\
&= \sum_{j=1}^{n} (w_{ij} - \delta_{ij}k_i)\boldsymbol{H}(\boldsymbol{x}_j)
\end{aligned} \tag{1.38}$$

where $k_i = \sum_j w_{ij}$ is the incoming degree of node i and δ_{ij} is the Kronecker delta. By introducing the Laplacian matrix, $\boldsymbol{L}$ with $L_{ij} = \delta_{ij}k_i - w_{ij}$ we obtain,

$$\boldsymbol{x}_i(t+1) = \boldsymbol{f}(\boldsymbol{x}_i(t)) - \sum_{j=1}^{n} L_{ij}\boldsymbol{H}(\boldsymbol{x}_j) \tag{1.39}$$

Then we can rewrite Eq. (1.39) in a compact form as,

$$\boldsymbol{X}(t+1) = \boldsymbol{F}(\boldsymbol{X}(t)) - (\boldsymbol{L} \otimes \boldsymbol{H})(\boldsymbol{X}(t)), \tag{1.40}$$

where $\boldsymbol{X} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n]^T$, $\boldsymbol{F}(\boldsymbol{X}) = [\boldsymbol{f}(\boldsymbol{x}_1), \cdots, \boldsymbol{f}(\boldsymbol{x}_n)]^T$ and $\otimes$ is the Kronecker product (Pereira *et al.*, 2013).

A synchronized oscillator network corresponds to a limit cycle of the network dynamics. As explained in the fixed points, limit cycles can be stable or unstable, and stability is determined similarly. We check the response of the system to small perturbations. Also, the mathematics to check the stability of the synchronous states is very similar to fixed points. Assume a dynamical system given by Eq. (1.37) and a periodic limit-cycle solution $\boldsymbol{x}_i(t) = \boldsymbol{s}(t)$ for all i. We can linearize the system if the solution is perturbed. Then, we expand the linearized solution as a combination of the eigenvectors of an appropriate matrix depending on the nature of the interaction between nodes. If that matrix is an adjacency matrix or Laplacian matrix, then the result based on the eigenvectors becomes a set of n decoupled systems. These isolated oscillating systems must be stable if the whole network is to be stable. To find a condition for the stability of the synchronized state, we need to define a *master stability function* using Lyapunov exponent (Newman, 2010).

1.5.3 Master stability function

Consider the function $\boldsymbol{h} : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^m$. We say that $\boldsymbol{h}$ is diffusive if

$$\boldsymbol{h}(x, x) = \boldsymbol{h}(0) = 0 \text{ and } \boldsymbol{h}(x, y) = -\boldsymbol{h}(y, x)$$

We again consider the model to a general diffusive coupling

$$\boldsymbol{x}_i(t+1) = \boldsymbol{f}(\boldsymbol{x}_i(t)) + \gamma \sum_{j=1}^{n} w_{ij} \boldsymbol{h}(\boldsymbol{x}_j, \boldsymbol{x}_i) \tag{1.41}$$

where γ is the coupling strength. We perform the analysis close to synchronization $\boldsymbol{x}_i = \boldsymbol{s} + \boldsymbol{\xi}_i$ so

$$\boldsymbol{h}(\boldsymbol{x}_j, \boldsymbol{x}_i) = \boldsymbol{h}(\boldsymbol{s} + \boldsymbol{\xi}_j, \boldsymbol{s} + \boldsymbol{\xi}_i) = \boldsymbol{h}(\boldsymbol{s}, \boldsymbol{s}) + D_1\boldsymbol{h}(\boldsymbol{s}, \boldsymbol{s})\boldsymbol{\xi}_j + D_2\boldsymbol{h}(\boldsymbol{s}, \boldsymbol{s})\boldsymbol{\xi}_i$$

but since the coupling is diffusive

$$D_2\boldsymbol{h}(\boldsymbol{s}, \boldsymbol{s}) = -D_1\boldsymbol{h}(\boldsymbol{s}, \boldsymbol{s})$$

we get

$$\boldsymbol{h}(\boldsymbol{x}_j, \boldsymbol{x}_i) = \boldsymbol{H}(\boldsymbol{s})(\boldsymbol{\xi}_j - \boldsymbol{\xi}_i) + \boldsymbol{R}(\boldsymbol{\xi}_i, \boldsymbol{\xi}_j)$$

where $\boldsymbol{H}(\boldsymbol{s}) = D_1\boldsymbol{h}(\boldsymbol{s},\boldsymbol{s})$ and $\boldsymbol{R}(\boldsymbol{\xi}_i,\boldsymbol{\xi}_j)$ contains quadratic terms. Then, the first variational equation about the synchronization manifold

$$\begin{aligned}\boldsymbol{\xi}_i(t+1) &= D\boldsymbol{f}(\boldsymbol{s}(t))\boldsymbol{\xi}_i(t) + \gamma\sum_{j=1}^{n} w_{ij}\boldsymbol{H}(\boldsymbol{s})(\boldsymbol{\xi}_j(t) - \boldsymbol{\xi}_i(t)) \quad (1.42)\\ &= D\boldsymbol{f}(\boldsymbol{s}(t))\boldsymbol{\xi}_i(t) - \gamma\boldsymbol{H}(\boldsymbol{s})\sum_{j=1}^{n} L_{ij}\boldsymbol{\xi}_j(t). \quad (1.43)\end{aligned}$$

All blocks have the same form which are different only by λ_i, the ith eigenvalue of $\boldsymbol{L}$. Then we obtain the parametric equation for the modes

$$\boldsymbol{u}(t+1) = [D\boldsymbol{f}(\boldsymbol{s}(t)) - \kappa\boldsymbol{H}(\boldsymbol{s}(t))]\boldsymbol{u}(t)$$

where $\kappa = \gamma\lambda_i$. By fixing κ, we compute the maximum Lyapunov exponent $\Lambda(\kappa)$ as

$$\|\boldsymbol{u}(t)\| \leq Ce^{\Lambda(\kappa)t}.$$

The map

$$\kappa \mapsto \Lambda(\kappa) \quad (1.44)$$

is called master stability function. Notice that if $\Lambda(\kappa) < 0$ when $\kappa > \gamma_c\lambda_2$ then $\|u\| \to 0$. For more details on the master stability function Eq. 1.44, see e.g., (Huang *et al.*, 2009; Schultz *et al.*, 2016).

2. THEORY AND MODELING

2.1 Modeling the Brain

Neural networks are described by chaotic isolated dynamics (Korn and Faure, 2003), weakly interacting nodes (Preyer and Butera, 2005) and interaction through scale-free type networks (Werner, 2010). To mimic neuron behavior, we use Rulkov maps,

$$\begin{aligned} u_i(t+1) &= \frac{\beta}{1+u_i(t)^2} + v_i(t) \\ v_i(t+1) &= v_i(t) - \nu u_i(t) - \sigma \end{aligned} \tag{2.1}$$

where the fast variable u_i is the membrane potential and the slow variable v_i is the ion concentration variation (Wang, Chen and Perc, 2011). The constant parameters $\beta = 4.1$ and $\nu = \sigma = 0.001$ are fixed for chaotic bursting dynamics (Rulkov, 2002). The regimes of neuronal behavior of the map for different β parameters are shown in Fig. 2.1. The mean duration of the bursts is very sensitive to β value.

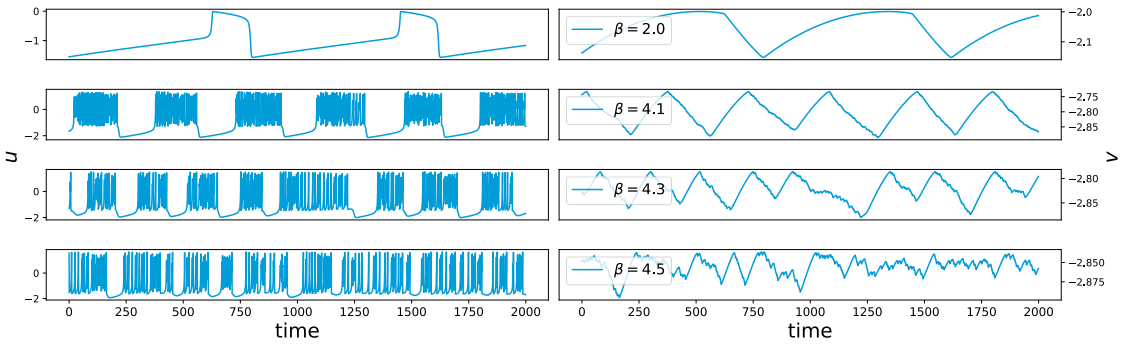

Figure 2.1 Bursting behavior generated by Rulkov map for $\beta = 2.0, 4.1, 4.3$ and 4.5. Left panel shows the u variable and right panel shows v variable. Isolated Rulkov maps are simulated 12000 time steps and discarded the first 10000 steps as transient.

Electrical coupling between neurons correspond u-coupled Rulkov maps diffusively by:

$$\mathbf{H}(\mathbf{x}_i, \mathbf{x}_j) = [h(u_i, u_j), 0] \tag{2.2}$$

where $\boldsymbol{x}_i \in \mathbb{R}^2$ with $h(u_i, u_j) = u_j - u_i$.

2.2 Reduction Theorem

A low-dimensional reduction of Eq. (1.37) is key for our network dynamics reconstruction approach. The reduction theorem applies a mean-field approach that relies on two main statements: (i) the statistical behavior of nodes' dynamics (frequency distribution of states) must be preserved and (ii) a large portion of nodes must be interacting with at least a few nodes in the network. These statements are satisfied with the given assumptions for the reduction theorem: *chaotic local dynamics* of expanding maps and *weak coupling* (to preserve the nodes' state distribution against fluctuations due to the interactions or external noise) and *scale-free networks* (most of the nodes have small degrees $k \sim n^{\epsilon}$, and some nodes are hubs with degrees $k \sim n^{\frac{1}{2}+\epsilon}$ where ϵ is an arbitrarily small number), which also mimic brain network dynamics. Using the theorem, the coupling term of Eq. (1) can be reduced as follows:

$$\begin{aligned}\sum_{j=1}^{n} w_{ij}\boldsymbol{H}(\boldsymbol{x}_i, \boldsymbol{x}_j) &\approx k_i \int \alpha\tilde{\boldsymbol{H}}(\boldsymbol{x}_i, \boldsymbol{x}_j) d\mu(\boldsymbol{x}_j) \\ &= k_i\alpha(\tilde{\boldsymbol{V}}(\boldsymbol{x}_i) + \tilde{C}) = k_i\boldsymbol{V}(\boldsymbol{x}_i) + C\end{aligned}$$

where $\boldsymbol{V}$ is the effective coupling function, $k_i = \sum_j w_{ij}$ is the incoming degree of node i, $\boldsymbol{H} = \alpha\tilde{\boldsymbol{H}}$, α is a multiplier for the coupling function, μ is a physical measure of the isolated dynamics, C is the integration constant and the integral takes into account the cumulative effect of interactions on node-i. As the coupling term is reduced as a function of an invariant measure μ, the reduction theorem works for a system in a steady state. Furthermore, to apply this mean-field approach-based reduction theorem, the statistical properties of individual dynamical systems must be preserved, which is satisfied by chaotic oscillators and weak coupling (Ref. (Pereira *et al.*, 2020)). Then Eq. (1.37) can be written as

$$\boldsymbol{x}_i(t+1) = \boldsymbol{f}(\boldsymbol{x}_i(t)) + k_i\boldsymbol{V}(\boldsymbol{x}_i(t)) + C + \boldsymbol{\kappa}_i(t) + \boldsymbol{\eta}_i(t) \tag{2.3}$$

where $\boldsymbol{\kappa}_i(t)$ is a small fluctuation for an interval of time that is exponentially large and depends on the state of neighbors of the ith node.

2.3 Network Reconstruction

2.3.1 Literature

The network dynamics reconstruction from data is a very active research field (Gao and Yan, 2022; Wang *et al.*, 2016; Nitzan *et al.*, 2017; Stankovski *et al.*, 2017; Timme and Casadiego, 2014; Casadiego *et al.*, 2017). Various methods were proposed to infer the connectivity matrix under some constraints, such as the need for a system to be at steady-state (Gardner *et al.*, 2003) or requiring prior knowledge about the dynamics (Timme, 2007; Shandilya and Timme, 2011; Yu *et al.*, 2006) or the coupling strength (Ren *et al.*, 2010). In addition to the studies that reveal the connectivity matrix by control signals or analytical solutions, statistical learning approaches such as compressed sensing were also introduced to learn entire unknown dynamics (Wang, Yang, Lai, Kovanis and Harrison, 2011; Wang *et al.*, n.d.), which also infers the connectivity structure. On the other hand, there are available regression-based approaches that can learn network topology using short time series (Casadiego *et al.*, 2017); however, it is impossible to detect the critical transitions with only the connectivity. However, statistical learning techniques are not extendable for large networks or require long time series measurements. A natural question is then whether revealing the network dynamics of weakly interacting chaotic oscillators would be possible using relatively short data without requiring knowledge of the system's nodal behavior and coupling scheme. This question is especially relevant in weak coupling regimes, in which the synchronization regime is unstable and the decay of correlation is exponential for chaotic oscillators, meaning that similarity measures cannot capture the interaction topology.

2.3.2 Proposed methodology

To learn isolated dynamics $\boldsymbol{f}$ and coupling function $\boldsymbol{H}$, we first need to classify nodes regarding their degrees. According to the reduction theorem, nodes with a similar in-degree must have a similar governing equation. To identify the governing

equations of each node independently from other nodes, we use sparse regression, particularly the Sparse Identification of Nonlinear Dynamical Systems (SINDy) technique (Brunton *et al.*, 2016). We denote the data collection of node-i by Eq.(1.37) as $\mathcal{X}_i = [\boldsymbol{x}_i(1), \ldots, \boldsymbol{x}_i(T-1)]^T$ and $\mathcal{X}_i' = [\boldsymbol{x}_i(2), \ldots, \boldsymbol{x}_i(T)]^T$. SINDy performs a sparse regression for the linear equation $\mathcal{X}_i' = \boldsymbol{\Psi}(\mathcal{X}_i)\boldsymbol{\Xi}_i$ to solve for $\boldsymbol{\Xi}_i = [\xi_i^1, \ldots, \xi_i^p]^T$, which is a vector of coefficients that defines the dynamics, where $\boldsymbol{\Psi} = [\psi_1, \ldots, \psi_p]$ represents a library of basis functions and is applied to $\boldsymbol{x}_i$ as

$$\boldsymbol{\Psi}(\mathcal{X}_i) = \overbrace{\begin{bmatrix} \psi_1(\boldsymbol{x}_i(1)) & \psi_2(\boldsymbol{x}_i(1)) & \ldots & \psi_p(\boldsymbol{x}_i(1)) \\ \psi_1(\boldsymbol{x}_i(2)) & \psi_2(\boldsymbol{x}_i(2)) & \ldots & \psi_p(\boldsymbol{x}_i(2)) \\ \vdots & \vdots & \ddots & \vdots \\ \psi_1(\boldsymbol{x}_i(T-1)) & \psi_2(\boldsymbol{x}_i(T-1)) & \ldots & \psi_p(\boldsymbol{x}_i(T-1)) \end{bmatrix}}^{\text{candidate functions of } \boldsymbol{x}} \Bigg\downarrow \text{time}$$

where p is the number of candidate functions in the library $\boldsymbol{\Psi}$. Sparse regression's goal is to determine the dynamics with a small number of functions in $\boldsymbol{\Psi}$ by finding active coefficients in $\boldsymbol{\Xi}_i$. Consequently, we obtain a predicted model for each node only using the associated node's own data, and we expect to learn similar models for the nodes with similar in-degree k_i. A distance matrix is obtained by normalized Euclidean distance to classify the predicted models $d_{ij} = (\Sigma_{k=1}^{p} \frac{1}{V_k} |\xi_i^k - \xi_j^k|^2)^{1/2}$, where $|\cdot|$ is absolute value, V_k is the variance of the predicted coefficients of the kth function in $\boldsymbol{\Psi}$. Assume $\boldsymbol{\Xi}_i$ and $\boldsymbol{\Xi}_j$ are two predicted models of nodes-i and j, which are presented as a linear combination of some functions within the library $\boldsymbol{\Psi}$. We get smaller d_{ij} for a similar pair of nodes i and j, while d_{ij} will be large for distinct nodes, such as a low-degree node and a hub. The histogram $P(D)$ is obtained by the row-sum of the distance matrix $D_i = \sum_j d_{ij}$, which provides an excellent classification for model similarities in terms of their degrees (Fig. 2.2 (a)). The low-degree nodes are expected to be located in the highest bin of the histogram since the network has many low-degree nodes. The models recovered for low-degree nodes are determined as our $\boldsymbol{f}$ with a negligible fluctuation $\boldsymbol{\kappa}_i$. Contrarily, the distance D_i is expected to be large for the hub nodes as they are the rarest. Therefore, hubs are located in the lowest bin of the histogram $P(D)$ (Fig. 2.2 (a)). Note that the success of the reconstruction depends on the separability of the low-degree nodes and hubs with respect to their degrees, which means the network topology plays an important role here (Eroglu *et al.*, 2020).

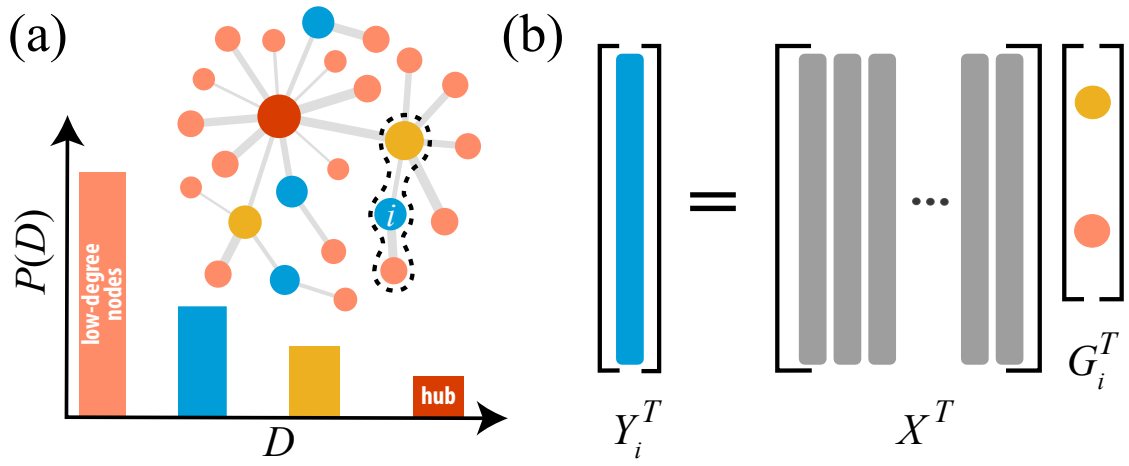

Figure 2.2 Illustration of the reconstruction scheme. (a) Performing sparse regression on each observation gives predicted models for each node. Nodes with the same in-degree are reconstructed with the same predicted models, which allow us to classify nodes concerning their in-degrees. As low-degree nodes are abundant and represent the isolated dynamics with a negligible noise, learning $\boldsymbol{f}$ is possible. Discarding the local dynamics, $\boldsymbol{f}$, from the hub's data gives the dominant coupling effect on the hub. Therefore, the coupling function $\boldsymbol{H}$ can be learned. (b) After learning $\boldsymbol{f}$ and $\boldsymbol{H}$, the problem is defined as a linear problem for each node by subtracting the local dynamics, which obtains the remaining interaction effect for each node. Sparse regression on the remaining interaction dynamics of node-i where $i = 1, \ldots, n$ entirely reconstructs the dynamical networks. Nonzero $\boldsymbol{G}_i^T$ elements are the incoming connections for ith node.

We obtain the cumulative coupling effect on the hub by discarding learned isolated dynamics contribution from the identified hub node's data as $\mathcal{X}'_h - \boldsymbol{f}(\mathcal{X}_h)$ where h denotes the hub node. We fit a function to the cumulative coupling effect on the hub and learn the coupling function $\boldsymbol{H}$ with a possible linear shift due to the integration constant C. The size of the linear shift can be easily estimated using $\boldsymbol{H}(\boldsymbol{0}) = \boldsymbol{0}$. Inferring the interaction function is vital to reveal the network (Liu and Barabási, 2016), and learning $\boldsymbol{H}$ from such reduced dynamics increases the feasibility of our approach. Defining $\boldsymbol{Y}(t) = \boldsymbol{X}(t+1) - \boldsymbol{F}(\boldsymbol{X}(t))$, Eq. (2.3) can be written as $\boldsymbol{Y} = \boldsymbol{G}\boldsymbol{X}$ where $\boldsymbol{G} = -(\boldsymbol{L} \otimes \boldsymbol{H})$. Finally, we complete the reconstruction by learning sparse matrix $\boldsymbol{G} \in \mathbb{R}^{mn \times mn}$, by solving the linear equation $\boldsymbol{Y}^T = \boldsymbol{X}^T \boldsymbol{G}^T$ using sparse regression, namely Least Absolute Shrinkage and Selection Operator (LASSO)(Tibshirani, 1996), as suggested in Ref. (Han *et al.*, 2015). LASSO adds the ℓ_1 regularization penalty to the least-squares loss function to find the sparse

coefficients (the links), and it is assessed as a compressed sensing approach (Candes *et al.*, 2006). Note that the linear equation can also be solved with ℓ_2-norm for long time series, however, as we are interested in short data (in the case of the length of the time series $T < mn$) the compressed sensing approach must be employed (Candes *et al.*, 2006). Consequently, we learn the connectivity matrices $\boldsymbol{G}$ and $\boldsymbol{L}$ as seen in Fig. 2.2 (b). It is also important to note that learning the equations of all nodes by a single sparse regression without the reduction theorem is only possible for relatively small networks. The library extension, due to network size, causes a statistically correlated data matrix that quickly fails on reconstruction (Novaes *et al.*, 2021; de Silva *et al.*, 2020).

2.3.3 Library of basis functions

Although prior knowledge about the dynamical system is always helpful in constructing a basis library, it is impossible to access the information for all cases. Nevertheless, many functions can be accurately estimated using high-order polynomials. For instance, in this work, we mainly studied with the Rulkov maps defined as follows,

$$\begin{aligned} u(t+1) &= \frac{\beta}{1+u(t)^2} + v(t) \\ v(t+1) &= v(t) - \nu u(t) - \sigma \end{aligned} \tag{2.4}$$

Using a basis library containing only polynomials, it is possible to reconstruct an imperfect model for the Rulkov map, which generates quite an accurate time series to the original one (Fig. 2.3). For the Rulkov map, only the non-polynomial term is the rational one, $\frac{1}{1+u^2}$, which is absent in the library. However, this rational term can be expanded in a power series as follows,

$$\frac{1}{1+x^2} = \sum_{n=0}^{\infty} (-x^2)^n = 1 - x^2 + x^4 - x^6 + x^8 + \cdots \tag{2.5}$$

therefore, it is possible to model the Rulkov map with polynomials roughly. As the power expansion's convergent interval is $[-1, 1]$, the predicted model diverges when the orbit is out of the given interval. However, such convergence problems can

be solved by normalizing the time series into the associated space. The interesting dynamics for our study is the bursting regime which can be modeled without the normalization Fig. 2.3 (b); therefore, we skip this step, but the model becomes very complicated with many higher-order polynomial terms.

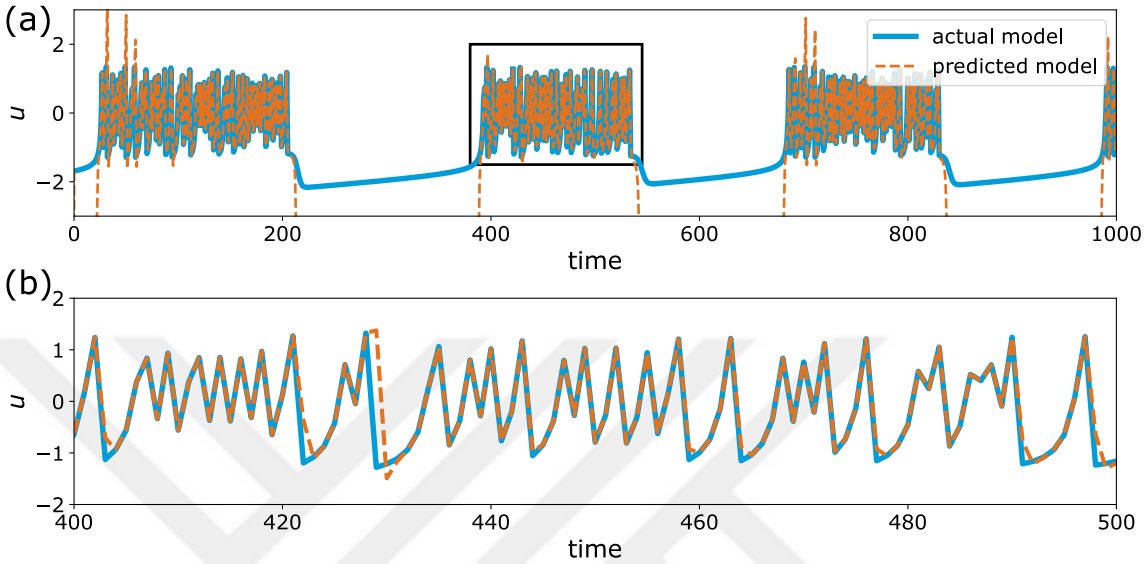

Figure 2.3 The predicted time series of a reconstructed Rulkov model using a candidate functions library containing only polynomials. a) predicted u-variable of the model b) a bursting regime package zoomed from the black rectangle in a).

In order to avoid complicated models containing many polynomial terms and possible inaccurate predictions, we prepared a standard library including common functions, such as geometric series and trigonometric functions. Although unreasonably increasing the number of functions in the library is undesired, adding such non polynomial terms generally terminates a large number of polynomials. Thus, our computational cost mostly becomes cheaper using the following standard candidate library:

$$\boldsymbol{\Psi} = \left[\mathbf{1}, \mathcal{P}_d(\boldsymbol{x}), \left\{\sin(i\boldsymbol{x}), \cos(i\boldsymbol{x})\right\}_{i=1}^{r}, \left\{\frac{1}{\boldsymbol{x}^i}, \frac{1}{1 \pm \boldsymbol{x}^i}, \frac{1}{(1 \pm \boldsymbol{x})^i}\right\}_{i=1}^{q}\right]$$

where d is the degree of the polynomials, and for instance $\mathcal{P}_2(\boldsymbol{x})$ is:

$$\mathcal{P}_2(\boldsymbol{x}) = [u, v, u^2, uv, v^2].$$

Our main strategy is to start $d = r = q = 1$, and increase them slowly to estimate the model with the minimum number of candidate functions. The first step of our

reconstruction algorithm aims to learn the local dynamics by classifying the inferred model equations due to their similarity. Here, we illustrate the reconstructed models on the mouse neocortex network using coupled Rulkov maps (Table 2.1) through a set of selected nodes' with respect to their degrees k_i ($k_0 = 0$, $k_{46} = k_{136} = 4$, and $k_{218} = 26$). As node-0 has no incoming links, we reconstructed only the Rulkov map dynamics. Node-46 has 4 incoming links, the inferred coefficients for it do not reflect the exact governing equation, but one can say that its governing equations are similar to node-136, which also has 4 incoming links. The most *distinct* model belongs to node-218, which is the hub of this particular network.

Table 2.1 Table for the inferred coefficients of the corresponding candidate functions for some of the selected nodes. We present only nonzero functions for any node.

node id	1	u	v	$\cdots$	$\cos(u)$	$\frac{1}{1-u}$	$\frac{1}{1+u^2}$
0	-0.000	-0.000	1.000	$\vdots$	0.000	-0.000	4.100
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
46	0.000	-0.015	1.005	$\vdots$	0.005	0.000	4.093
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
136	-0.000	-0.015	1.006	$\vdots$	0.000	-0.000	4.101
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
218	-0.714	0.098	0.783	$\vdots$	0.016	0.001	4.075

The first step of the reconstruction algorithm is to classify nodes, which uses a distance between nodes' coefficients. We define the distance metric as,

$$d_{ij} = (\Sigma_{k=1}^{p} \frac{1}{V_k} |\xi_i^k - \xi_j^k|^2)^{1/2} \tag{2.6}$$

where $|\cdot|$ is absolute value, V_k is the variance of the predicted coefficients of the k-th function in $\mathbf{\Psi}$. As an example, we compute the pairwise distances of the

learnt models given in Table 2.1: As numerically shown in Table 2.2, d_{ij} is small for

Table 2.2 Pairwise distances of the predicted models for some of the selected nodes.

node id	0	46	136	218
0	0.000	2.830	2.105	18.351
46	2.830	0.000	1.925	16.808
136	2.105	1.925	0.000	16.979
218	18.351	16.808	16.979	0.000

the nodes with the same or similar degrees, such as the distance is 1.925 between 46 ($k_{46} = 4$) and 136 ($k_{136} = 4$), while the distance is large for different predicted models, such as the distance is 18.351 between 46 ($k_{46} = 4$) and 218 ($k_{218} = 26$).

2.3.4 A toy example

We present a step-by-step reconstruction algorithm using weakly coupled chaotic maps on a directed and weighted network of size $n = 20$, shown in Fig. 2.4(a). The network is heterogeneous; the low-degree nodes are abundant, and the hub is the rarest, as illustrated in the network's in-degree distribution in Fig. 2.4(b). The network has Rulkov maps interacting with each other through their u-components diffusively:

$$\begin{aligned} u_i(t+1) &= \frac{\beta}{1+u_i(t)^2} + v_i(t) - \sum_{j=1}^{n} L_{ij} u_j(t) \\ v_i(t+1) &= u_i(t) - \nu u_i(t) - \sigma \end{aligned} \tag{2.7}$$

where the constant parameters $\beta = 4.1$ and $\nu = \sigma = 0.001$ are fixed for chaotic bursting dynamics. We first simulate the system 15000 time steps and discard the first 14000 steps as a transient. We should note that the local dynamics (Rulkov map), the coupling function (u-coupling) and the connectivity matrix (L_{ij}) are all unknowns during the procedure; however, we present every piece of information in this toy model to show the method rigorously.

Fig. 2.4(c) shows the 2-dimensional data coming from the hub and one of the low

degree nodes. We present the return maps of the same nodes in Fig. 2.4(d), the dispersion of the hub node due to the dominant coupling effect can be seen. On the other hand, we compute the pairwise Pearson correlation coefficients s_{ij} between the time series, see Fig. 2.4(e). Then we obtain the histogram $P(S)$, Fig. 2.4(f), by the row-sum of the correlation matrix in Fig. 2.4(e). Given the chaotic nature of the Rulkov maps, correlations between time series does not give any information about the network connectivity. Therefore, it is impossible to identify a low-degree node or the hub. The reconstruction procedure is given by the following steps:

1. **Classification of the nodes.** The separability of low-degree nodes and hubs is crucial since the reconstruction procedure is not applicable if we cannot classify which nodes are low-degree or hubs at the beginning of the reconstruction recipe. The classification leads us to learn the isolated dynamics $\boldsymbol{f}$ and the coupling function $\boldsymbol{H}$ as follows: we, first, assume that the coupling is *weak*, meaning that if we classify the low-degree nodes, we can consider their dynamics as local dynamics $\boldsymbol{f}$ with some small fluctuations. If we classify which node is the hub and $\boldsymbol{f}$, then we can learn the coupling function $\boldsymbol{H}$ by discarding $\boldsymbol{f}(\boldsymbol{x}_h(t))$ from the hub's time series $\boldsymbol{x}_h(t+1)$ using the cumulative effect of interactions. Therefore, our first aim is to classify the nodes regarding their degrees.

   For the classification, we first learn a governing equation for each time series using sparse regression methods with a basis composed of candidate functions. After obtaining the n predicted models, we measure the difference between the models using normalized Euclidean distance between the coefficients of predicted models, d_{ij}. A distance matrix $D = [d_{ij}]$ summarizes the mismatches between the predicted models. Fig. 2.4(g). The row-sum of the distance matrix $D_i = \sum_j d_{ij}$ gives a histogram $P(D)$, and each bin of the histogram contains nodes with a similar degree (Fig. 2.4(h)). It does also not contain the exact in-degree distribution of the network (Fig. 2.4(b)), however, allows for the low-degree and hub node identification.

   It is also important to note that weak coupling is also crucial to avoid possible

synchronization in the network dynamics. If the system is synchronized, the time series will be identical; in other words, all the predicted models will be identical. This will make the classification impossible. Therefore, the weak coupling is quite crucial for this algorithm to work as desired.

2. **Learning the local dynamics.** To mimic brain network dynamics, we assume that the connectivity matrix $\boldsymbol{W} = [w_{ij}]$ represents a scale-free network whose degree distribution follows a power law; in other words, most of the nodes have small degrees $k \sim n^{\epsilon}$, and some nodes are hubs with degrees $k \sim n^{\frac{1}{2}+\epsilon}$ where ϵ is an arbitrarily small number. Therefore, the highest bin of the histogram (the most abundant frequencies) contains the low-degree nodes' models since there are many low-degree nodes (the orange bar in Fig. 2.4 (h)). As it is likely to have nodes having incoming links, $k_i > 0$, in the set of low-degree nodes, the models can contain some small fluctuations. Thus, we average the predicted coefficients of the low-degree nodes' models to infer the local dynamics as accurately as possible.
3. **Learning the coupling function.** Using the scale-free network topology assumption as in the previous section, we understand that the lowest bin of the histogram (the rarest frequencies) contains the most different models, which belong to the hubs (the blue bar in Fig. 2.4(h)). Then discarding the local dynamics of the hub ($\boldsymbol{f}(\boldsymbol{x}_h(t))$ from the hub measurement ($\boldsymbol{x}_h(t+1)$) gives the dominant coupling effect, which we aim to learn. Fig. 2.4(i) presents the remaining coupling effect. We fit a function to the data and learn an approximate coupling function plus an integration constant arising from the reduction theorem. To be more precise with the integration constant, for instance, if the diffusive coupling function is given as $h(x, y) = \phi(y) - \phi(x)$ then the effective coupling V for the expending maps is $V = \int h(x, y) d\mu(y) = -\phi(x) + C$ where C is the integration constant, which we call a *possible linear shift* for the numerically recovered effective coupling V. This is easy to estimate from the fitted coupling (Fig. 2.4 (i)) since we assume that $\phi(0) = 0$. Therefore, the appearing linear shift is the integration constant C, and its value can vary for different dynamical networks. Furthermore, the effective coupling is an ap-

proximation, which becomes exact when the network model is considered as a directed star graph with the link directions from low-degree nodes to the hub at the thermodynamical limits as $n \to \infty$. We have an approximated effective coupling for tree-like graphs, as scale-free graphs, under our assumptions related to node degrees (mentioned above). The size of the bounded fluctuation term κ_i in the reduction (Eq. (2.3)) can be considered as the measure of approximation.

Here, the weak coupling and chaotic dynamics assumptions take an important role again for more general networks to preserve the statistical behavior of the nodes' dynamics. Although using chaotic oscillators helps to keep the state distribution of the states stable, this property can be destroyed by a large coupling. Therefore, if the network is denser, then the coupling constant should be scaled to a smaller coupling strength for our reduction theorem to work. For instance, if the hubs are rare in a scale-free network, the coupling strength must be scaled by $n^{1/2}$. If all the nodes are hubs, in other words, if the network is all-to-all connected, then the coupling strength must be scaled by n. In our work, we scaled the weights in $\boldsymbol{W}$ with $n^{1/2}$, as we considered scale-free networks (containing many low-degree nodes and rare hubs) since the work focuses on reconstructing brain networks. It is also important to remind that, for the node classification step, the synchronization of the nodes disallows the reconstruction since the machine learning-based methods cannot distinguish the data source between synchronized oscillators. Consequently, the chaotic oscillators also play an important role here since if we have periodic and identical oscillator networks, any positive coupling strength can synchronize the model, which is undesired for the reconstruction part. Therefore, the weak coupling strength and the chaotic dynamical regime are crucial assumptions for the theory and also for the numerical steps to learn the coupling function $\boldsymbol{H}$.

4. **Learning the connectivity matrix.** After learning the functions $\boldsymbol{f}$ and $\boldsymbol{H}$, we define $\boldsymbol{Y} = \boldsymbol{X}(t+1) - \boldsymbol{F}(\boldsymbol{X}(t))$, so Eq. ((2.3)) can be written as $\boldsymbol{Y} = \boldsymbol{G}\boldsymbol{X}$ where $\boldsymbol{G} = -(\boldsymbol{L} \otimes \boldsymbol{H})$. Therefore, learning the links becomes a regression

problem by solving the linear equation $\boldsymbol{Y}^T = \boldsymbol{X}^T\boldsymbol{G}^T$. Here we employ a compressed sensing approach using ℓ_1-norm, called LASSO, to solve the equation since we are interested in reconstructing large network dynamics from short data. In other words, we aim to solve the problem when the length of the time series $< m \times n$ where m is the dimension of the local dynamics and n is the system size. The problems under this condition are called underdetermined regression problems. Finally, we identify the exact Laplacian matrix of the network by solving the linear equation (Fig. 1 (j)).

Note that, in the previous steps of the procedure, the potential interaction functions of two nodes (or higher-order interactions if the hypernetworks are interested) are not involved in the basis library. Meaning that the size of the library is not grown exponentially. Therefore, the algorithm does not require a longer time series to reconstruct the network dynamics as the network size, n, is increasing. In this last step, we also only sparsely solve a linear equation where $\boldsymbol{G} \in \mathbb{R}^{mn \times mn}$, which is a square matrix.

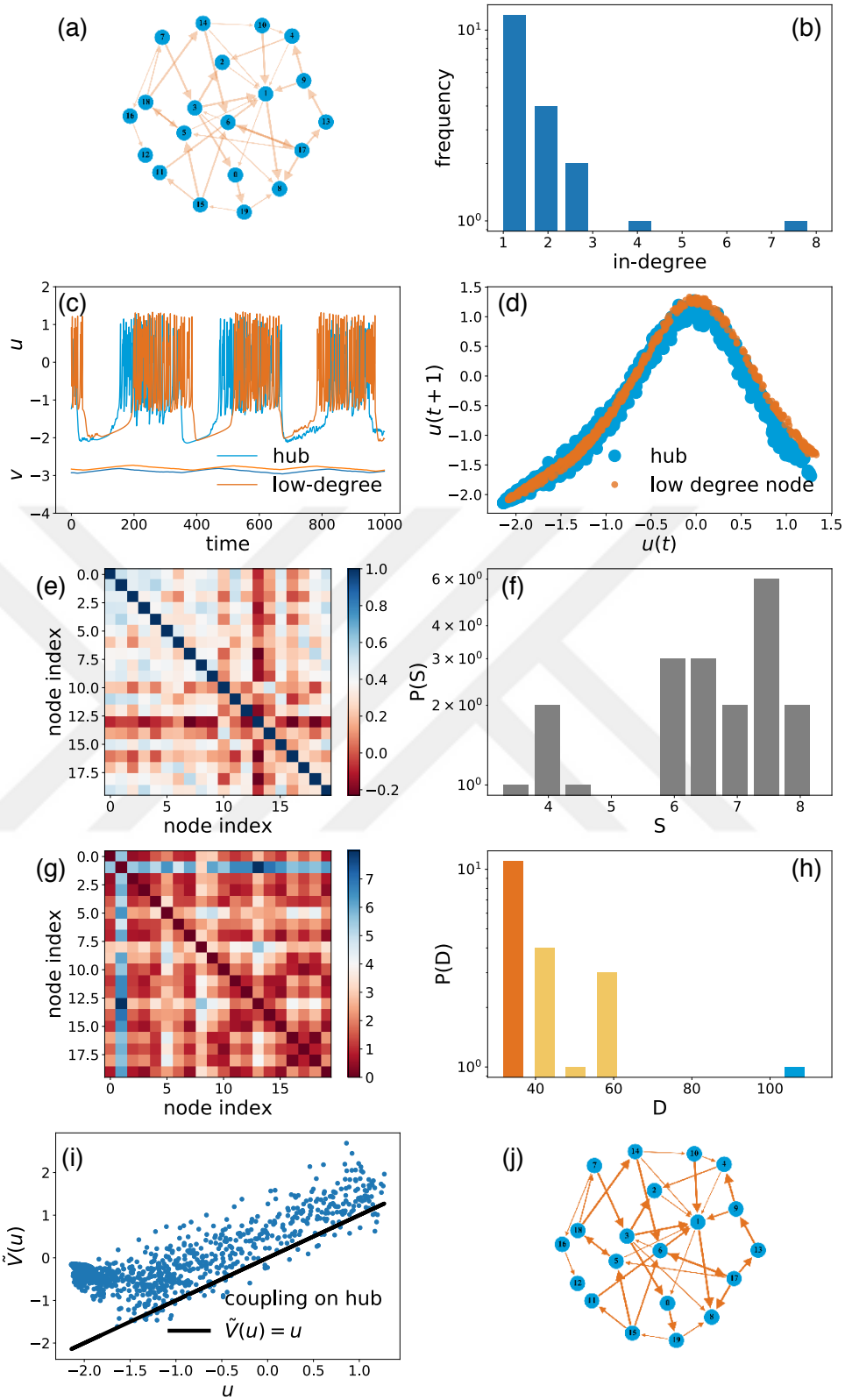

Figure 2.4 A scheme for the proposed reconstruction methodology on a small network of size $n = 20$. As the network is unknown initially, its links were illustrated indistinctly.

2.4 Synthetical Networks

The scale-free networks are generated using the algorithm in Ref. (Bollobás *et al.*, 2003), and weights are assigned uniformly from the interval $[0.8, 1.2]$. The algorithm creates undesired self-loops and multiple edges, so first we remove them. The algorithm we use to generate directed scale-free networks has three probabilistic parameters: α and γ are the probabilities of adding a new node connected to an existing node chosen randomly according to the in-degree and out-degree distribution, respectively. β is the probability of adding an edge between two existing nodes. By different parameter settings sparsity level of the network may be tuned.

2.5 Real Networks

2.5.1 Mouse neo-cortex network

We choose a neuronal connectivity consisting 1029 nodes representing a small volume of a young adult mouse neocortex (Kasthuri *et al.*, 2015). The network has 1700 synapses (links) which are manually labeled from a image data-set and the data can be found in the public repository (Vogelstein *et al.*, 2018). The network is directed and contains multi-edges between some nodes. We considered the number of these multi edges between nodes as weights. Furthermore, the network contains 19 disconnected parts, we removed them and kept only the giant component. The resulting weighted and directed network has $n = 987$ nodes and $m = 1536$ links, which is illustrated in Fig. 2.5 (a) with its in-degree distribution in Fig. 2.5 (b). We downscale entries of the connectivity matrix by a small constant 0.1, and then normalize by in-degree of hub, k_h, so that the network is not synchronized.

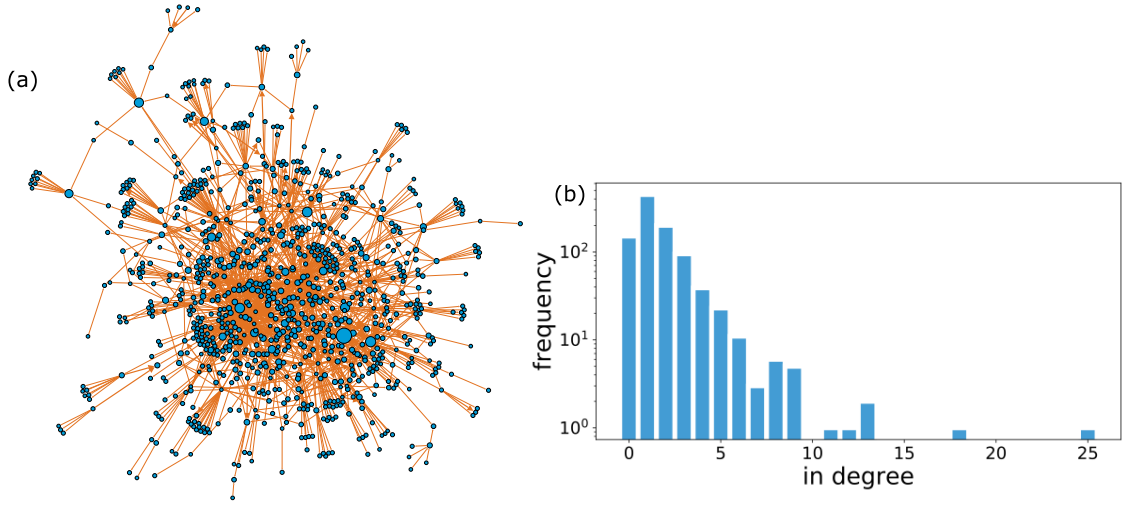

Figure 2.5 (a) Pre-processed directed and weighted mouse neo-cortex network. The sizes of the nodes denote their in-coming degrees and the thickness of the arrows denote their weights. (b) In-degree distribution of the associated real network.

2.5.2 Macaque monkey visual cortex network

We tried our procedure on a real monkey visual cortex network that is not a scale-free type (Markov *et al.*, 2014; Vogelstein *et al.*, 2018). The network consists of 91 nodes and 581 undirected and unweighted edges, which is illustrated in Fig. 2.6(a) with its degree distribution in Fig. 2.6(b). It is impossible to classify nodes by degree in a network with such a degree distribution, failure to classify nodes results in an unable to learn local dynamics and identify the hub. So, we assume we know the local dynamics and which node is the hub. When we continue with this preliminary information, the reduction theorem allows us to learn the coupling function smoothly and see full reconstruction.

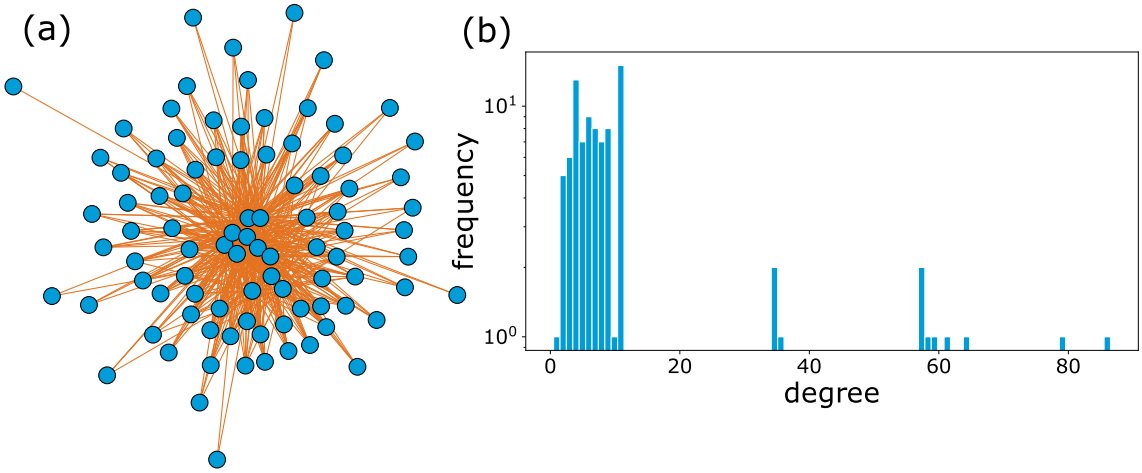

Figure 2.6 (a) Undirected and unweighted macaque monkey visual-cortex network. (b) Degree distribution of the associated real network.

3. RESULTS

3.1 Experiments

We use two standard indices to assess the reconstruction performance of our procedure: the fraction of the false negatives out of the positives (FNR) and the fraction of false positives out of the negatives (FPR). These metrics correspond to two types of errors in network reconstruction: missing a link where a link indeed exists (false negatives) or assigning a link between two nodes when they are not connected (false positives):

$$FNR = \frac{FN}{P} \text{ and } FPR = \frac{FP}{N}$$

where FN evaluates errors on the non-zero terms (P) of the correct matrix (underestimation) and FP evaluates errors of the zero terms (N) (overestimation). We count FN and FP links by $\sum_{i,j} \Theta(|L_{ij} - \hat{L}_{ij}| > \epsilon)$ to assess the accuracy of the link strengths where $\hat{L}_{ij}$ is the predicted Laplacian and ϵ is a tolerance value set as 10 times smaller than the smallest nonzero value in $\boldsymbol{L}$. We use a fixed tolerance value as $\epsilon = 0.0001$ for all analyses.

3.1.1 Noise-free case

Following the reconstruction scheme, the nodes are classified using the similarity histogram Fig. 3.1(a) for $\eta_0 = 0$, while the pairwise Pearson correlations do not show any information about the degrees (inset in Fig. 3.1(a)). The difference between return maps of a low-degree node and the hub is illustrated in Fig. 3.1(b). In our reconstruction scheme, we assume that a low-degree node can be taken as an isolated node to estimate local dynamics $\boldsymbol{f}$. Effective coupling $\boldsymbol{V}(\boldsymbol{x})$ is found approximately $[0.1(u+1), 0]$, meaning that α is 0.1 and the linear shift is 1 on u-variable due to $\boldsymbol{H}(\boldsymbol{0}) = \boldsymbol{0}$ (Fig. 3.1(c)). Finally, we learn the network topology by solving the linear

equation $\boldsymbol{Y} = \boldsymbol{G}\boldsymbol{X}$ using the learned $\boldsymbol{f}$ and $\tilde{\boldsymbol{H}}$. We measure the reconstruction error using the fraction of the false negatives (positives) out of the positives (negatives), FNR (FPR). The FNR (FPR) equals 0 for perfect reconstruction. Here, we use the ground truth Laplacian matrix to assess the accuracy of the reconstruction. When the ground truth is not available, the learned model can be evaluated by the cross-validation techniques. The reconstruction error is found to be almost zero for a data length larger than $T > 200$ (Fig. 3.1(d)).

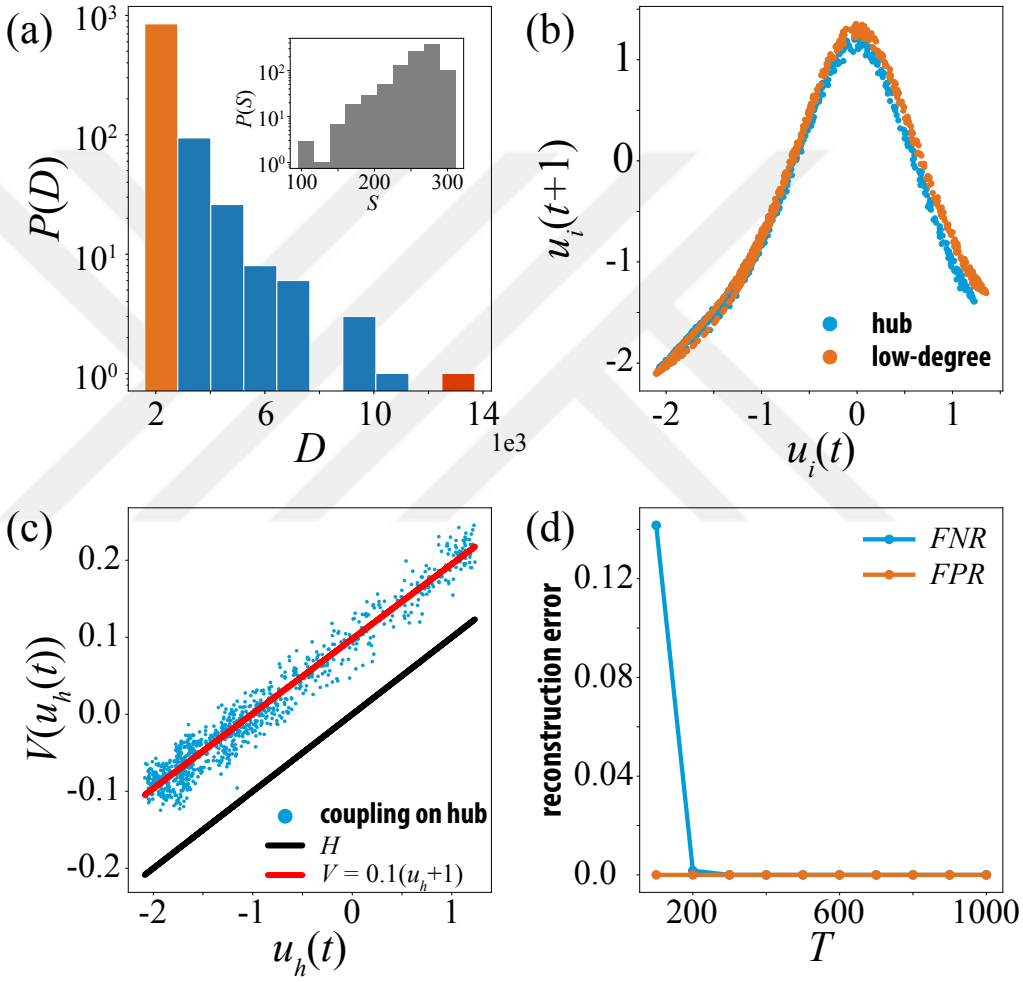

Figure 3.1 Reconstruction procedure for weakly electrically coupled Rulkov maps on a real mouse neocortex network. (a) Node-similarity histogram determines the low-degree nodes (orange bar) and the hub (red bar). Inset: Histogram $P(S)$ presents the correlations between the original time series. It is impossible to infer the connectivity structure from the correlations due to the chaotic nature of the Rulkov maps. (b) The return maps of a low-degree node and the hub are slightly different due to weak coupling effect. (c) The effective coupling, $V(u)$, shifts through the horizontal direction due to the integral constant 1. (d) FNR for different lengths of time series. FPR are zero for all time series lengths.

We compare the return maps of a low-degree node and a hub node's dynamics with an isolated Rulkov map under the effect of weak coupling in Fig 3.2.

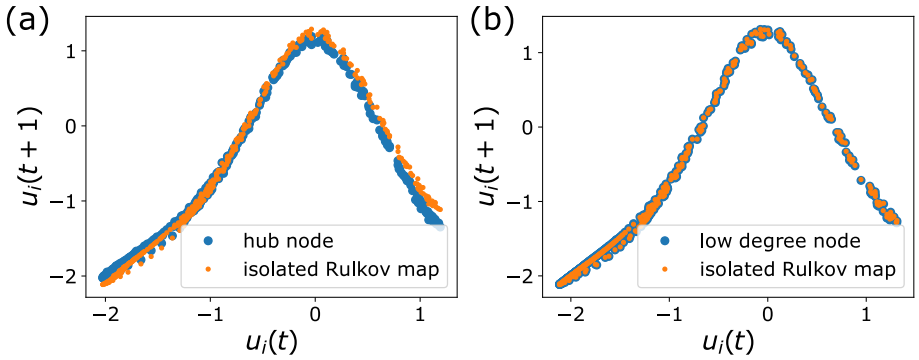

Figure 3.2 a) The return map of the hub's data and isolated Rulkov map to show the weak coupling effect. b) The return map of one of the low-degree nodes shows that these nodes oscillate almost with the isolated local dynamics.

Furthermore, a systematical evaluation of our approach according to penalty terms and time series lengths is performed. When the number of nodes n times the dimension of the local dynamics m exceeds the time series length $nm > T$, it corresponds to an underdetermined linear problem. Even for short data ($T \approx 200$), a successful reconstruction is possible with a small penalty term when $mn = 1974$ (Fig. 3.3). Note that if the problem is overdetermined $nm < T$, then ℓ_2-norm regression can be used for faster computations.

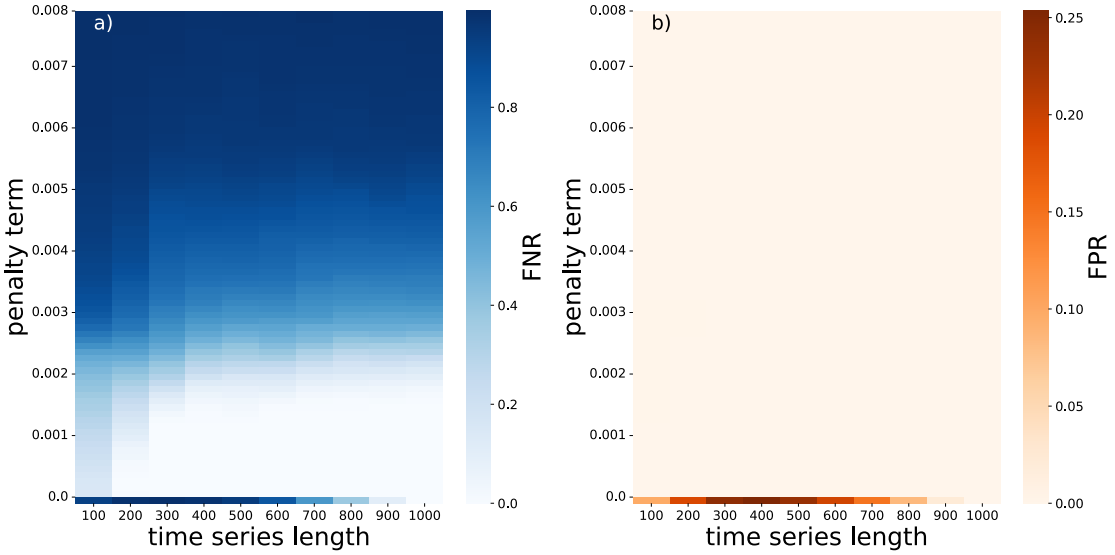

Figure 3.3 (a) The reconstruction performance for different time series lengths and a series of penalty terms as FNR and (b) as FPR.

,

3.1.2 Noise effect

To measure the robustness of our methodology against noise, we systematically perform the reconstruction procedure for various η_0 values (Eq. (1.37)) on synthetic as well as real mouse neocortex networks. The reconstruction approach is robust to small noise intensities for the real-world example (Fig. 3.4). The average robustness of the reconstruction procedure over 50 different directed and weighted random scale-free networks is given in Fig. 3.5.

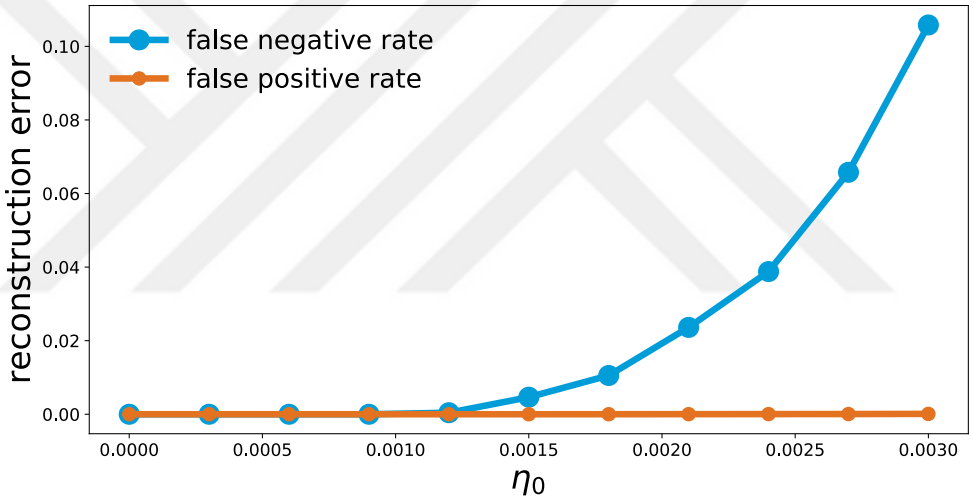

Figure 3.4 Noise effect on reconstruction performance on real network as FNR and FPR.

As the system size grows, the reconstruction performance reduces with respect to FNR for increasing noise intensity η_0 (Fig. 3.5(a)), since the noise becomes more dominant than the weak coupling, which prohibits learning $\boldsymbol{H}$ using the coupling effect. As similar to real-world application, FPR results are also negligibly small for the noise induced reconstruction case (Fig. 3.5(b)).

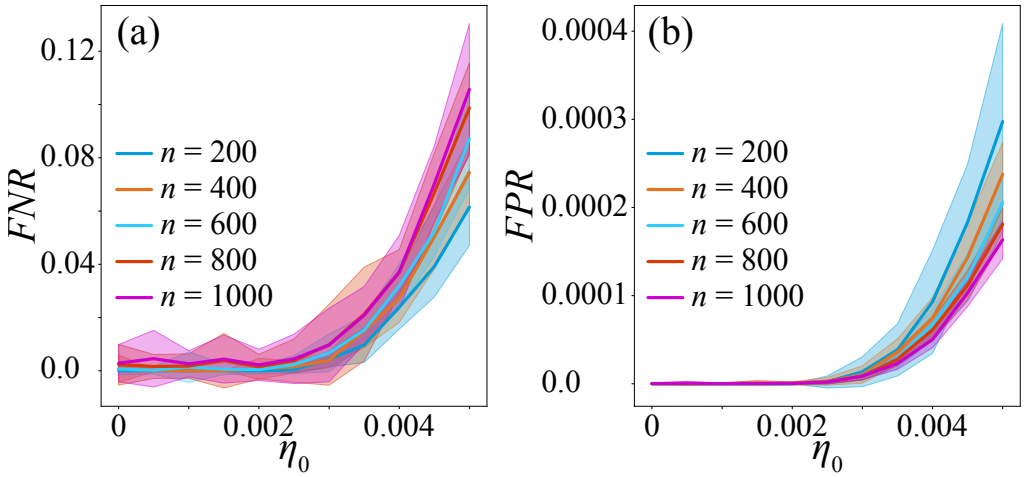

Figure 3.5 Average noise effect on reconstruction performance illustrated using (a) FNR and (b) FPR for 50 realizations of simulations using random scale-free networks of system sizes $n = 200, 400, 600,$ 800 and 1000. Time series length is fixed as 500 during all the simulations. The shaded regions represent the corresponding standard deviations.

3.1.3 Comparison between sparse regression methods

We compare two optimizers to perform sparse regression; LASSO and STLS.
In the final step of our reconstruction approach, we find a sparse solution for $\boldsymbol{w}$, it corresponds to $\boldsymbol{L}$ in our case. From the algorithmic point of view, fine-tuning of the hyper-parameters, penalty term for LASSO and sparsity parameter for STLS, is essential to improve the accuracy. These two parameters are not equivalent, however we can make a similar interpretation between them (Mehta *et al.*, 2019). The length of the time series determines the problem type since our network size is fixed in the experiments. The time series length lower than the network size ($T < nm = 987 \times 2 = 1974$) correspond to an under-determined case for the real network we use.

Fig. 3.6 presents the reconstruction performance for different time series lengths and two hyperparameters as FNR and FPR. We compare FNR values to evaluate the performance of LASSO and STLS in Fig. 3.6(a) and Fig. 3.6(c), since FPR values are always very small. The results show that LASSO overcomes STLS for shorter time series. Since the ℓ_1-norm penalized solution, is a generalization of compressive

sensing approach, it provides a unique solution for underdetermined cases (Donoho, 2006). On the other hand, if there is long enough data, STLS becomes a fast alternative to the LASSO from an algorithmic point of view and works very well for our problem. We took one under-determined case ($T = 300$) to compare the performance of two different sparsity promoting methods in Fig. 3.7. If we have limited data, we see full reconstruction by LASSO but not by STLS.

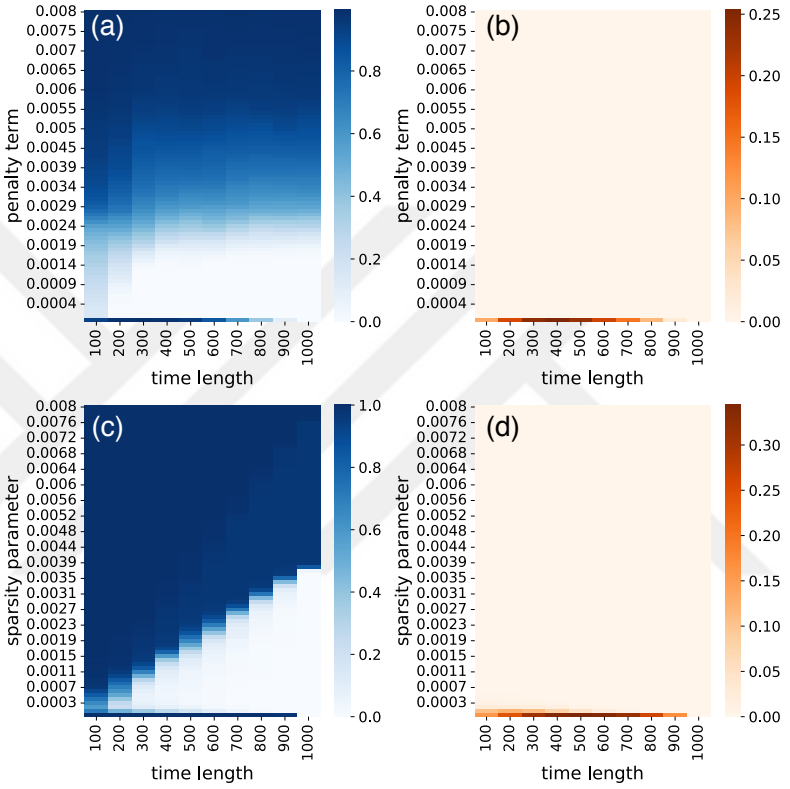

Figure 3.6 (a) FNR and (b) FPR with respect to a list of penalty terms for LASSO. (c) FNR and (d) FPR with respect to a list of sparsity parameters for STLS. 10 different lengths of time series are used to compare the performance.

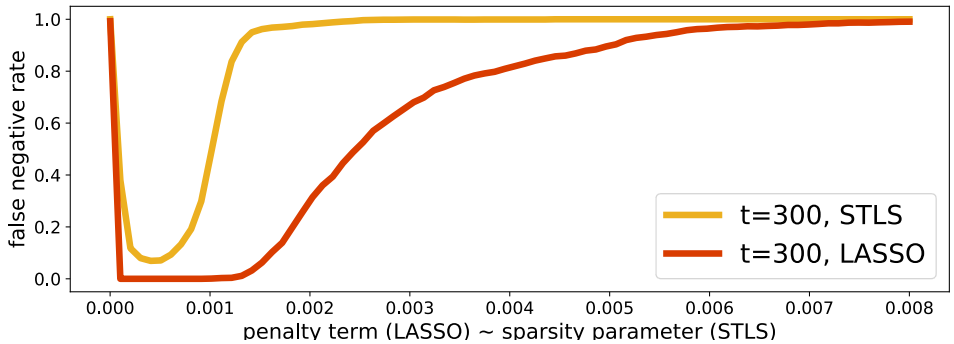

Figure 3.7 FNR with respect to a list of penalty term for Lasso and sparsity parameter for STLS. ($T = 300$) case is used as a short time series example. There is no *full reconstruction* region for STLS.

3.1.4 Sparsity effect

In addition to the lack of sufficient data, the sparsity condition of the Laplacian matrix should be satisfied to solve the linear regression problem within the compressed sensing framework. We present results in this direction based on synthetical networks. We use two different parameter setting as $[\alpha = 0.41, \gamma = 0.05, \beta = 0.54]$ and $[\alpha = 0.2, \gamma = 0.5, \beta = 0.3]$ to obtain Laplacian matrices at different sparsity levels. The former (default) setting's power exponent is approximately 2 and generates denser networks, while the latter's power exponent is approximately 1 and generates sparser networks. As seen in Fig. 3.8, we see full reconstruction in sparser networks since the Laplacian matrix meets the sparsity condition for the compressive sensing approach (Candes *et al.*, 2006).

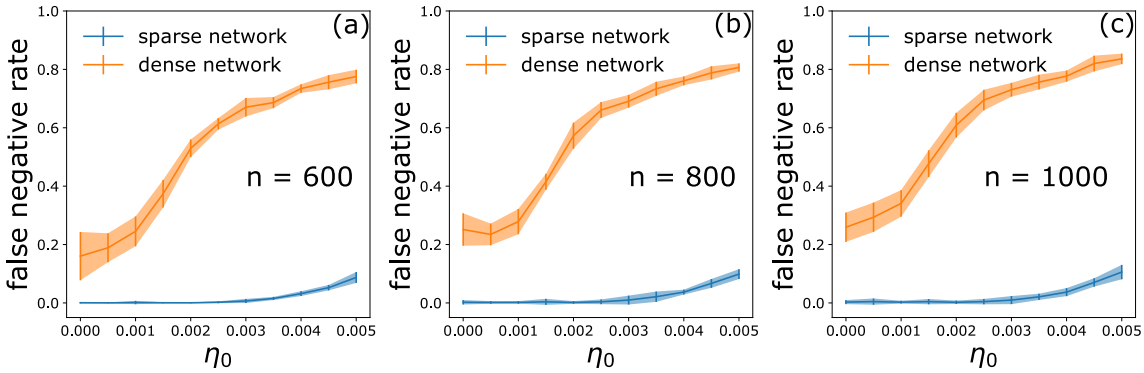

Figure 3.8 FNR against noise for sparse and dense networks of sizes (a) $n = 600$, (b) $n = 800$ and (c) $n = 1000$. All points show an average error over 10 different realizations of scale free networks and shaded regions present standard deviations. Time series length is 500, which corresponds to underdetermined cases for all three network sizes.

3.1.5 Cross validation of inferred Laplacian matrices

In the main results, we use the ground-truth Laplacian matrix to show the performance of the reconstruction for various hyper-parameters and data lengths. However, it is unlikely to know the true matrix to evaluate the methodology's success in real-world problems. Therefore, it is important to assess the reconstruction outcome when the real matrix is absent using a cross-validation technique. To illustrate the cross-validation approach, we obtained *predicted time series* of length 500 using inferred Laplacian matrices for various hyper-parameters (penalty term) $\lambda = 0.008, 0.003$ and 0.0003. Out of those three parameters, the outcome for $\lambda = 0.0003$ gives the best-fit, Fig. 3.9(a).

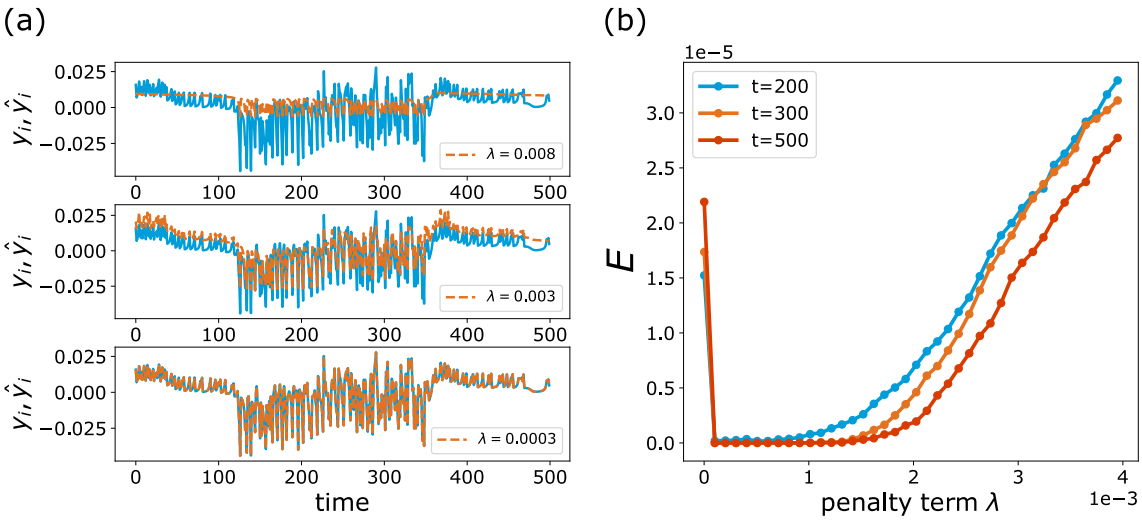

Figure 3.9 a) Comparison between measurement of a node and test time series. Each λ parameter corresponds to different connectivity matrices. We simulate the inferred models and obtain test time series. b) Cross validation of the inferred time series based on obtained connectivity matrices for a series of penalty terms.

To find the best hyper-parameters in large network dynamics, we use an error function

$$E = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{1}{T}\sum_{t=1}^{T}\left(y_{i,t} - \hat{y}_{i,t}\right)^2\right) \tag{3.1}$$

where $y_i = x_i(t+1) - f(x_i)$ and $\hat{y}$ denotes the predicted one. Then we optimize the hyper-parameters, which minimize the error. The parameters yielding the most accurate results can be used as the optimal hyper-parameter (Fig. 3.9(b)).

3.1.6 Other dynamical systems

We provide reconstruction analysis for Hénon map and Tinkerbell map.

Hénon map.

u-component coupling. We use Hénon map as $\boldsymbol{f}$ in Eq. (1.39), a discrete-time dynamical system exhibit chaotic behavior:

$$\begin{aligned} u_i(t+1) &= 1 - \alpha u_i(t)^2 + v_i(t) - \sum_{j=1}^{n} L_{ij} u_j(t) \\ v_i(t+1) &= \beta u_i(t) \end{aligned} \tag{3.2}$$

where $\alpha = 1.4$ and $\beta = 0.3$ are fixed during the simulations. Hénon maps are coupled through their u-components weakly. Initial positions are taken from the interval $[0, 0.1]$ randomly uniformly for each map. We iterate over 11000 steps and discard the first 10000 steps as a transient. Our procedure reveals the local dynamics as *Hénon map*, interaction function as *diffusive coupling* for $u-$component, and finally, the correct Laplacian matrix from time series data.

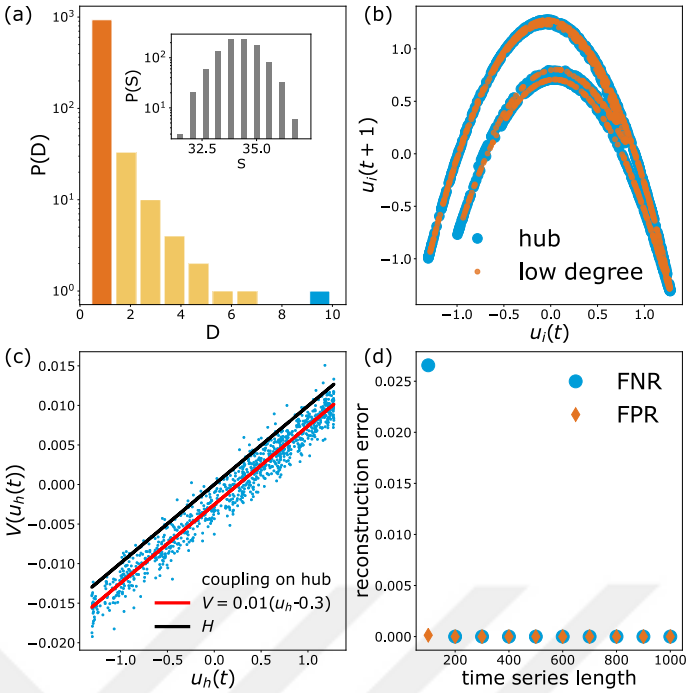

Figure 3.10 Reconstruction procedure for weakly u-coupled Hénon maps on a real network. First, we learn a model for each node's time series data to classify them. (a) The histogram $P(D)$ based on the pairwise Euclidean distance matrix. While the highest bin of the histogram gives us $\boldsymbol{f}$, the lowest bin identifies the hub. The inset histogram emphasizes that the pairwise correlations of the time series observations do not provide any information about the network connectivity due to the chaotic dynamics. (b) The return maps of a low degree node and hub show the coupling effect slightly due to weak coupling. In (c), we plot the dominant coupling effect on the hub to see the model of $\boldsymbol{H}$. (d) shows FNR and FPR for different lengths of time series. 200 data points are enough for full reconstruction of the network dynamics with 987 nodes.

u-component coupling with sinusoidal function. We use the same $\boldsymbol{f}$ but different coupling function $\boldsymbol{H}$ in this example:

$$\begin{aligned} u_i(t+1) &= 1 - \alpha u_i(t)^2 + v_i(t) - \sum_{j=1}^{n} L_{ij} \sin(2\pi u_j(t)) \\ v_i(t+1) &= \beta u_i(t) \end{aligned} \tag{3.3}$$

Initial positions are taken from the interval $[0, 0.01]$ randomly uniformly for each map. We iterate over 12000 steps and discard the first 10000 steps as a transient. Fig. 3.11 shows the reconstruction results of sine-coupled Hénon maps. We simply added trigonometric functions to the candidate function library in the first step of our procedure and fully reconstructed the network connections by learning the local

dynamics as *Hénon map* and the coupling function as *sine function*.

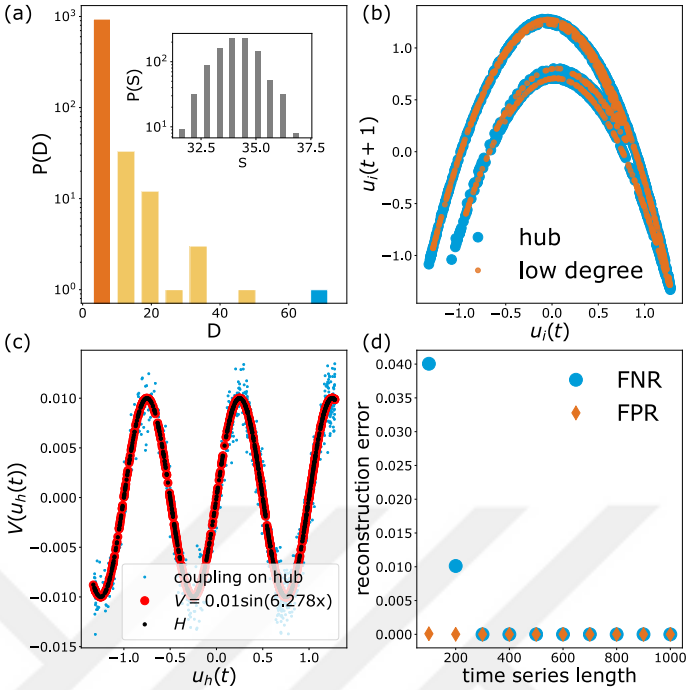

Figure 3.11 Reconstruction procedure for weakly $\sin u$-coupled Hénon maps on a real network. First, we learn a model for each node's time series data to classify them. (a) The histogram $P(D)$ based on the pairwise Euclidean distance matrix. While the highest bin of the histogram gives us $\boldsymbol{f}$, the lowest bin identifies the hub. The inset histogram emphasizes that the pairwise correlations of the time series observations do not provide any information about the network connectivity due to the chaotic dynamics. (b) The return maps of a low degree node and hub show the coupling effect slightly due to weak coupling. In (c), we plot the the dominant coupling effect on the hub to see the model of $\boldsymbol{H}$. (d) shows FNR and FPR for different lengths of time series. 300 data points are enough for full reconstruction of the network dynamics with 987 nodes.

Tinkerbell Map. Tinkerbell map coupled through their u-component diffusively is given by:

$$\begin{aligned} u_i(t+1) &= u_i(t)^2 - v_i(t)^2 + au_i(t) + bv_i(t) - \sum_{j=1}^{n} L_{ij}u_j(t) \\ v_i(t+1) &= 2u_i(t)v_i(t) + cu_i(t) + dv_i(t) \end{aligned} \tag{3.4}$$

where $a = 0.9, b = -0.6013, c = 2.0$ and $d = 0.5$ are fixed during the simulations to ensure a fully chaotic regime. Data is generated from u-coupled Tinkerbell maps by iterating over 11000 steps and dropping the first 10000 time steps as a transient. We

successfully reveal the local dynamics of the Tinkerbell map, interaction function and finally the Laplacian matrix from time series.

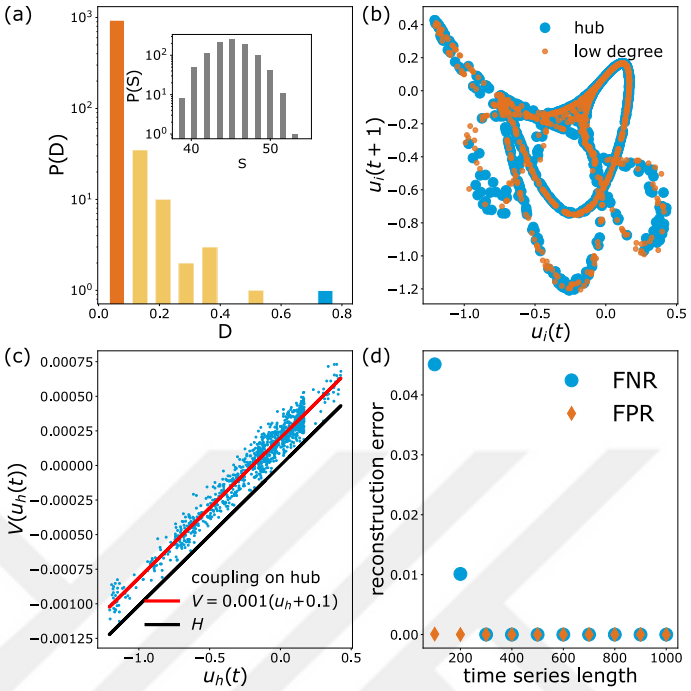

Figure 3.12 Reconstruction procedure for weakly u-coupled Tinkerbell maps on a real network. First, we learn a model for each node's time series data to classify them. (a) The histogram $P(D)$ based on the pairwise Euclidean distance matrix. While the highest bin of the histogram gives us $\boldsymbol{f}$, the lowest bin identifies the hub. The inset histogram emphasizes that the pairwise correlations of the time series observations do not provide any information about the network connectivity due to the chaotic dynamics. (b) The return maps of a low degree node and hub show the coupling effect slightly due to weak coupling. In (c), we plot the dominant coupling effect on the hub to see the model of $\boldsymbol{H}$. (d) shows FNR and FPR for different lengths of time series. 300 data points are enough for full reconstruction of the network dynamics with 987 nodes.

As seen in figures 3.10(c), 3.11(c) and 3.12(c), the horizontal shift between the fitted curve on the data (red) and the predicted H curve (black) varies for each dynamical system. This observation is compatible with the reduction theorem, which is analytically proven only for expanding maps (Eroglu *et al.*, 2020). We show that even if the theorem is not rigorously proven for the Rulkov map, Hénon map or Tinkerbell map, the methodology accurately works. For the coupling, $H(x, y) = y - x$, $v(x) = \int H(x, y)dm(y)$ is found as $-x$ plus a constant. These constants are found as 1 for coupled Rulkov maps, -0.3 for coupled Hénon maps and 0.1

for coupled Tinkerbell maps. For sinusoidal coupling, $H(x, y) = \sin 2\pi y - \sin 2\pi x$, $v(x) = \int H(x, y)dm(y)$ is found as $-\sin 2\pi x$ plus a constant and it is 0 for sin-coupled Hénon maps.

3.2 Prediction of Critical Transitions

Although we recover the true network structure, the reduction theory approximates the isolated dynamics $\boldsymbol{f}$ and coupling function $\boldsymbol{H}$ with small bounded fluctuations. As the local dynamics is chaotic and we consider possible noise effect, time evolution forecasting of the given system can diverge from the original system. However, as we recovered the network dynamics with high accuracy for the given data as,

$$\boldsymbol{X}(t+1) = \boldsymbol{F}(\boldsymbol{X}(t)) + \gamma \boldsymbol{G}\boldsymbol{X}(t), \tag{3.5}$$

where γ is the coupling control parameter and initially it is $\gamma = 1$ for the reconstructed model. Then, it is possible to detect critical coupling strength factor γ_c to predict the emergent behavior of the dynamical network by fully analytical techniques if the reconstructed coupling function is identity matrix (Eroglu *et al.*, 2017). For general coupling functions, master stability function (Pecora and Carroll, 1998) or connection graph method (Belykh *et al.*, 2004) can be performed for the detection.

3.3 Why is the Reduction Theorem Crucial to Reconstruct Large Networks?

A fully-algorithmic approach is to apply any sparse optimization technique directly to the entire multi-variable time series. Only up to a certain size networks can be recovered by this method. By the reduction theorem, we learn local dynamics from low degree nodes and interaction dynamics from the hub. We are not looking for a sparse linear combination of functions within a massive library of all possible functions. Therefore, there is no limit on network size in our approach. In this section, we will show this with a simple example:

Assume we have a network with 5 nodes. If node 1 has links coming from the nodes

0, 4 and 5 as seen in Fig. 3.13, then the dynamics of node 1 is written simply in Laplacian matrix form:

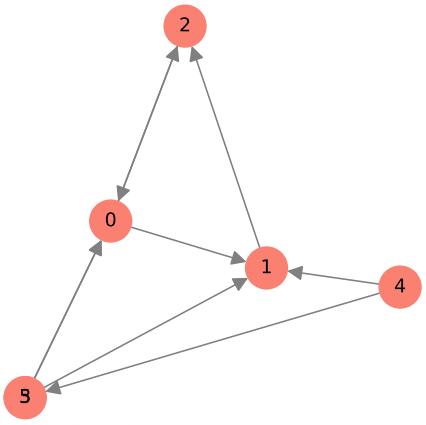

Figure 3.13 5-nodes network as an example.

$$\boldsymbol{x}_1(t+1) = \boldsymbol{f}(\boldsymbol{x}_1(t)) + 3\boldsymbol{H}(\boldsymbol{x}_1(t)) - \boldsymbol{H}(\boldsymbol{x}_0(t)) - \boldsymbol{H}(\boldsymbol{x}_4(t)) - \boldsymbol{H}(\boldsymbol{x}_5(t)) \qquad (3.6)$$

where $\boldsymbol{x}_i \in \mathbb{R}^m$. Eq.(3.6) is in the form of a linear combination of the connections, allowing any sparse regression algorithm to capture the parameters correctly in case of a well-defined basis library. For one single equation, it seems pretty feasible, but for a 5-nodes network, each has m components, we have $5m$ equations to be discovered. Suppose we set the basis composed of polynomials up to second-degree and all quadratic combinations of all variables due to a piece of prior knowledge about the coupling function. This example setting has 9 functions for $m = 3$. In that case, our basis includes 45 features for this particular example. If we do not have prior knowledge, we use an extensive library that includes more candidate functions. We performed experiments in this direction to see the limitations. The results showed that a purely algorithmic equation-based learning approach fails as the network size, that is, the basis grows. Results of Novaes *et al.* validated our conclusions, they showed that adding new functions to the basis due to increase of network size (up to 20-nodes) effects badly the reconstruction (Novaes *et al.*, 2021). Furthermore, in the final sparse regression step, our problem turns into a linear equation as $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}$, where $\boldsymbol{y} \in \mathbb{R}^T$, $\boldsymbol{A} \in \mathbb{R}^{T \times n}$ and $\boldsymbol{x} \in \mathbb{R}^n$. Here, $\boldsymbol{y}$ is the

data, and the length of the data is T, n is the network size, and we focus on the underdetermined cases ($T < n$) since data availability is always limited. From the perspective of compressed sensing, the measurement matrix $\boldsymbol{A}$ should satisfy some conditions to converge the sparsest solution for $\boldsymbol{x}$ by ℓ_1-norm. One of these conditions can be summarized as the time series length scales with the network size, so if the measurement matrix has more columns, longer measurements are needed. It is given by the relation:

$$T \approx \mathcal{O}(Klog(n/K))$$

where K is the network's sparsity measure (number of nonzero entries) (Brunton and Kutz, 2019). If the number of columns is determined by the basis functions that include all possible pairwise interactions, not the network size, this means the need for a longer time series. On the other hand, the uncertainty of the interaction function is an obstacle to inferring the exact connection matrix (Liu and Barabási, 2016). Finding $\boldsymbol{H}$ and therefore reducing the dimension of the optimization problem for full reconstruction broadens the applicability and distinguishes our work.

3.4 Sparse Recovery on a Real Data

The authors in Ref. (Cherevko *et al.*, 2016) proposed a generalized Van der Pol-Duffing equation to model blood flow in the cerebral vessels. They obtained patient-specific clinical experimental data during the neurosurgical operations in Meshalkin Research Institute of Circulation Pathology (Novosibirsk, Russia). The measurement contains velocity and pressure data obtained at different stages of the operations and belongs to two different malformations: arterial aneurysms (AA) and arteriovenous malformations (AVM).

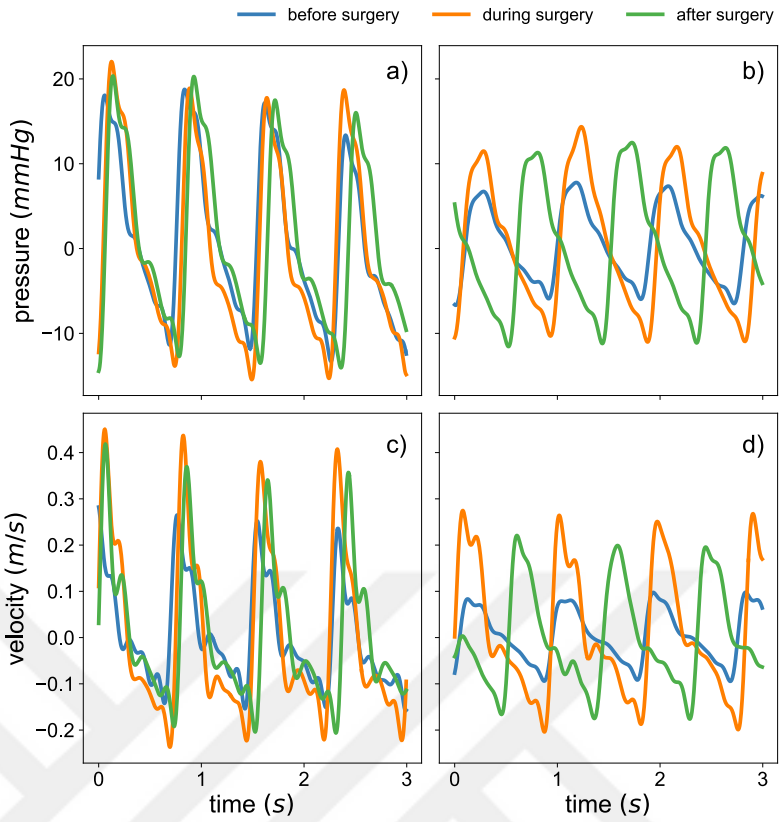

Figure 3.14 The upper panels show blood flow pressure (a) and velocity (b) of one of AA patients, while the lower panels show one of AVM patient's pressure (c) and velocity (d)

They used coefficient inverse methods to fit their model on the data, and these methods are computationally expensive in terms of real-time clinical data processing. We use SINDy (Brunton *et al.*, 2016) to fit a simplified linear model (3.7) for each clinical patient-specific pressure data and to speed up the model construction process. Now, the aim is to find an equation of form $\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t))$. We have the linear equation $\dot{\boldsymbol{X}} = \boldsymbol{\Theta}(\boldsymbol{X})\boldsymbol{\Xi}$ to solve for $\boldsymbol{\Xi}$. Since the measurement $\dot{X}$ is not available, we approximate them numerically for the left-hand side of the equation.

$$\ddot{p}(t) + a\,\dot{p}(t) + b\,p = \epsilon u(t) \tag{3.7}$$

Fig. 3.15 displays the predicted pressure obtained by the proposed model in P-V plane with clinical pressure for one AA and AVM patient. It showed that the sparse recovery approach could predict a pressure model which replicated the real pressure observations before, during and after surgery.

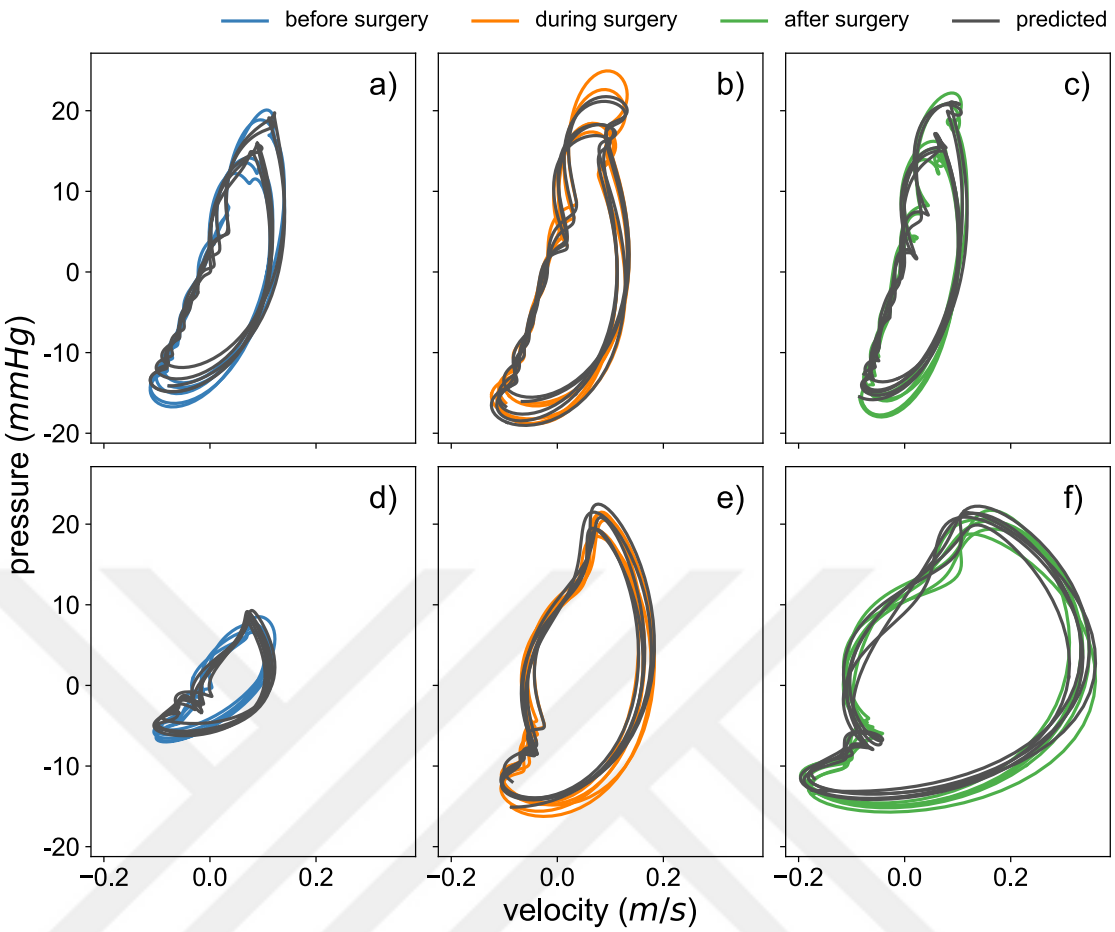

Figure 3.15 Pressure - velocity plane for one patient's before (blue), during (orange) and after (green) surgery data: (a-c) patient with AA; (d-f) patient with AVM. Dark curves present the simulated pressure data from the proposed model. Predicted data show a counterclockwise current which validates the circulation in arteries.

The mean squared error between the ground-truth time series and the predicted time series is a common measure to assess the model accuracy as:

$$\text{mse}(p, \hat{p}) = \frac{1}{n_s} \sum_{i=0}^{n_s-1} (p - \hat{p}_i)^2, \tag{3.8}$$

where n_s is the number of samples, p is the ground-truth pressure and $\hat{p}$ is the modeled pressure. Four different training sample sizes, the number of observations in a measurement, are used for evaluating the forecasting performance of the predicted model. As seen in Fig. 3.16(a), the test sample size for each case is almost the same.

SINDy is applied to the *training set*, the inferred models are simulated forward in time and obtain *predicted* series. Fig. 3.16(b) shows the mean squared error calculated between the predicted series and test sets. The length of the training set affects the forecasting accuracy as expected.

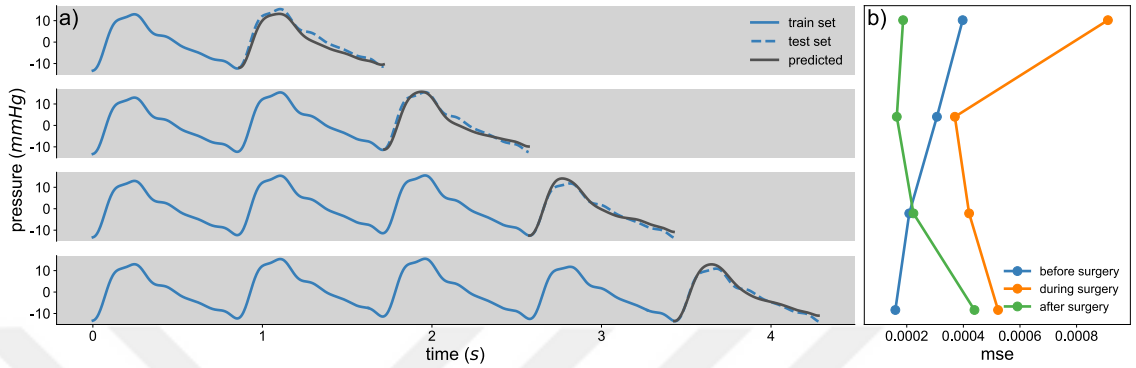

Figure 3.16 Forecasting performance of the proposed model on AA patients.

We will split the data into two parts to discover models independently and see how unique the predicted coefficients are.

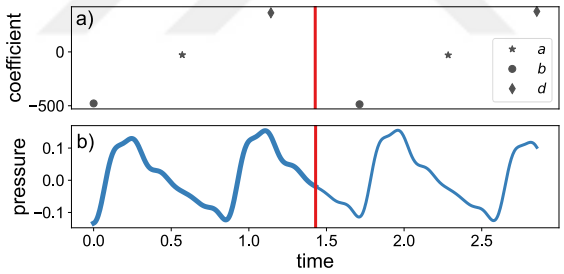

Figure 3.17 a) Predicted coefficients of the learnt model on AA patients. b) Splitted pressure data collected before surgery of one of the AA patients.

Furthermore, we test whether the fitted parameter values of Eq. (3.7), regarded as a data-dimensionality reduction, can be used for automatic classification of blood-flow pathologies. To this end we apply the logistic regression to patients data to discriminate flow with AA, AVM, and in surgically treated vessels. We encode these

groups as quantitative response variables:

$$Y = \begin{cases} 0 & \text{if AA patient} \\ 1 & \text{if AVM patient} \\ 2 & \text{if treated patient} \end{cases} \tag{3.9}$$

Fig. 3.18 shows the results based on different subsets of 3 features. We apply logistic regression on randomly selected 80% of the patients and test our model on the remaining 20%. The average accuracy over 100 random selections are 0.75 ± 0.19 when a and b are used, 0.48 ± 0.23 when b and d are used and 0.36 ± 0.21 when a and d are used in the training.

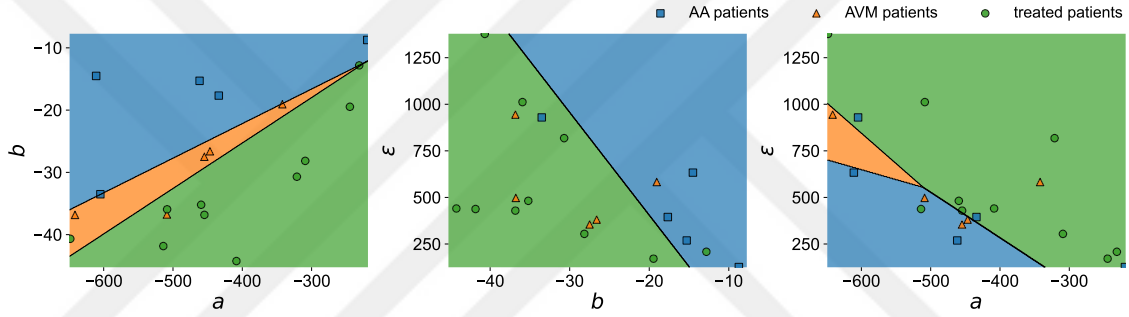

Figure 3.18 Data contains model coefficients based on 5 AA patients before surgery, 5 AVM patients before surgery and 10 AA and AVM patients after surgery data. The black lines are the decision boundaries.

Fig. 3.19 shows the results based on the training with 3 features. The average accuracy over 100 random selections are 0.73 ± 0.2 when a, b and d are used together.

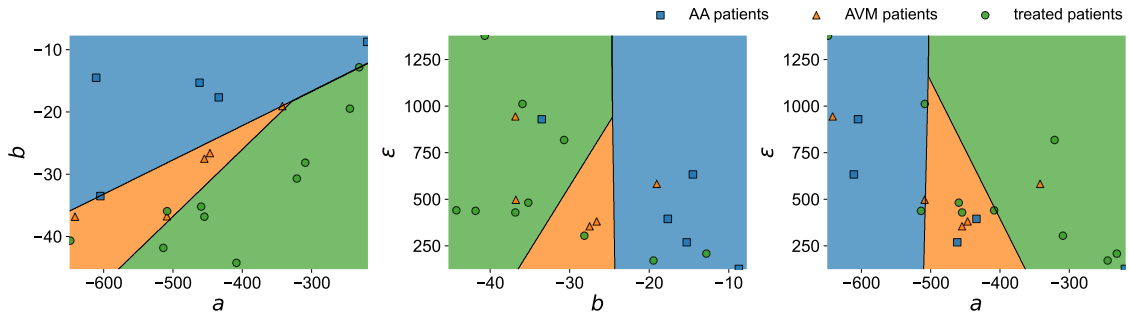

Figure 3.19 Data contains model coefficients based on 5 AA patients before surgery, 5 AVM patients before surgery and 10 AA and AVM patients after surgery data. Each plot shows two of three features by fixing the third feature value to its mean value.

The overall goal is to build a computational framework for real-time monitoring during cardiovascular surgeries. One of the main objectives of our work is to use a technique which would overcome the previous approach in terms of model-building time. SINDy is fast to discover a governing model for the clinical data, and we showed that the proposed model is robust. Moreover, we used the patient-specific model parameters to classify the two malformations before and after surgery. 73% accuracy in the classification task shows that SINDy as a model discovery technique can be used as a dimensionality reduction method for time series.

4. CONCLUSIONS

The network is fully recovered by our approach in the setting where $\boldsymbol{f}$ is chaotic, the network is scale-free and the coupling is weak. Because of the weak coupling between nodes and the chaotic nature of the local dynamics, the correlation between measured time series decays exponentially; therefore, it is impossible to reconstruct such complex systems by conventional methods. Cutting-edge autonomous statistical learning techniques also fail when the network size is large. The key idea in our procedure is splitting the model equation into parts by reduction theorem and inferring each unknown ($\boldsymbol{f}, \boldsymbol{H}$ and $\boldsymbol{W}$) one by one using sparse recovery. Although the reduction theorem is not established for the general chaotic discrete maps, we showed its validity on various maps. Our approach guarantees the full reconstruction for the noise-free case and small noise intensity even for relatively short time series with no limitations on the network size. However, the quality of reconstruction decreases for increasing noise and the destructive effect of the noise also increases with increasing system size. Finally, obtaining the network dynamics allows one to predict the emergent behavior under parameter changes. The ability to detect such transitions is crucial for applications such as a transition to collective behavior in the brain network, which can lead to undesired implications. Thus, it is desirable to put forward precautionary norms to avert potential disasters.

Continuous dynamical systems on complex networks remain open questions for our approach. To perform sparse regression independently on each node's measurement, we set up the left-hand side of the equations one step further from the time series in the case of the discrete dynamical system. However, we would put the derivatives of the state variables in continuous systems. Numerical derivatives may give poor results. Also, hidden variables in the data and needing access to all nodes to get measurements still need to be solved for our approach.

BIBLIOGRAPHY

Alligood, K., Sauer, T. and Yorke, J. (2000), *Chaos: An Introduction to Dynamical Systems*, Textbooks in Mathematical Sciences, Springer New York.

Barabási, A.-L. and Pósfai, M. (2016), *Network science*, Cambridge University Press, Cambridge.

Barzel, B. and Barabási, A. L. (2013), 'Universality in network dynamics', *Nature Physics* **9**, 673–681.

Belykh, V. N., Belykh, I. V. and Hasler, M. (2004), 'Connection graph stability method for synchronized coupled chaotic systems', *Physica D: Nonlinear Phenomena* **195**, 159–187.

Bollobás, B., Borgs, C., Chayes, J. and Riordan, O. (2003), Directed scale-free graphs, *in* 'Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms', SODA '03, Society for Industrial and Applied Mathematics, p. 132–139.

Brunton, S. L. and Kutz, J. N. (2019), *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*, Cambridge University Press.

Brunton, S. L., Proctor, J. L., Kutz, J. N. and Bialek, W. (2016), 'Discovering governing equations from data by sparse identification of nonlinear dynamical systems', *Proceedings of the National Academy of Sciences* **113**, 3932–3937.

Candes, E. J., Romberg, J. K. and Tao, T. (2006), 'Stable signal recovery from incomplete and inaccurate measurements', *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences* **59**(8), 1207–1223.

Casadiego, J., Nitzan, M., Hallerberg, S. and Timme, M. (2017), 'Model-free inference of direct network interactions from nonlinear collective dynamics', *Nature communications* **8**(1), 1–10.

Cherevko, A. A., Mikhaylova, A. V., Chupakhin, A. P., Ufimtseva, I. V., Krivoshapkin, A. L. and Orlov, K. Y. (2016), 'Relaxation oscillation model of hemodynamic parameters in the cerebral vessels', *Journal of Physics: Conference Series* **722**, 012045.

de Silva, B. M., Champion, K., Quade, M., Loiseau, J.-C., Kutz, J. N. and Brunton, S. L. (2020), 'Pysindy: A python package for the sparse identification of nonlinear dynamical systems from data', *Journal of Open Source Software* **5**(49), 2104.

Donoho, D. (2006), 'Compressed sensing', *IEEE Transactions on Information Theory* **52**(4), 1289–1306.

Dörfler, F., Chertkov, M. and Bullo, F. (2013), 'Synchronization in complex oscillator networks and smart grids', *Proceedings of the National Academy of Sciences* **110**(6), 2005–2010.

Eroglu, D., Lamb, J. S. and Pereira, T. (2017), 'Synchronisation of chaos and its applications', *Contemporary Physics* **58**(3), 207–243.

Eroglu, D., Tanzi, M., van Strien, S. and Pereira, T. (2020), 'Revealing dynamics, communities, and criticality from data', *Physical Review X* **10**, 1–13.

Gao, T.-T. and Yan, G. (2022), 'Autonomous inference of complex network dynamics from incomplete and noisy data', *Nature Computational Science* **2**, 160–168.

Gardner, T. S., Bernardo, D. D., Lorenz, D. and Collins, J. J. (2003), 'Inferring genetic networks and identifying compound mode of action via expression profiling', *Science* **301**, 102–105.

Gauss, C. F. (1823), *Theoria combinationis observationum erroribus minimis obnoxiae*, Vol. 1, Henricus Dieterich.

Gleick, J. (1987), *Chaos: Making a new science*, Viking, New York.

Han, X., Shen, Z., Wang, W. X. and Di, Z. (2015), 'Robust reconstruction of complex networks from sparse data', *Physical Review Letters* **114**, 028701.

Hastie, T., Tibshirani, R. and Friedman, J. (2009), *The Elements of Statistical Learning*, 2 edn, Springer.

Huang, L., Chen, Q., Lai, Y.-C. and Pecora, L. M. (2009), 'Generic behavior of master-stability functions in coupled nonlinear dynamical systems', *Physical Review E* **80**, 036204.

Izhikevich, E. M. (2007), *Dynamical systems in neuroscience*, MIT press.

James, G., Witten, D., Hastie, T. and Tibshirani, R. (2014), *An Introduction to Statistical Learning: With Applications in R*, Springer Publishing Company, Incorporated.

Kasthuri, N., Hayworth, K. J., Berger, D. R., Schalek, R. L., Conchello, J. A., Knowles-Barley, S., Lee, D., Vázquez-Reina, A., Kaynig, V. and *et al.*, T. R. J. (2015), 'Saturated reconstruction of a volume of neocortex', *Cell* **162**, 648–661.

Korn, H. and Faure, P. (2003), 'Is there chaos in the brain? ii. experimental evidence and related models', *Comptes Rendus - Biologies* **326**, 787–840.

Legendre, A. M. (1805), *Nouvelles methodes pour la determination des orbites des cometes*, F. Didot.

Liu, Y. Y. and Barabási, A. L. (2016), 'Control principles of complex systems', *Reviews of Modern Physics* **88**, 035006.

Lloyd, A. L. (1995), 'The coupled logistic map: a simple model for the effects of spatial heterogeneity on population dynamics', *Journal of Theoretical Biology* **173**(3), 217–230.

Lorenz, E. N. (1963), 'Deterministic nonperiodic flow', *Journal of the Atmospheric Sciences* **20**, 130–141.

Markov, N. T., Vezoli, J., Chameau, P., Falchier, A., Quilodran, R., Huissoud, C., Lamy, C., Misery, P., Giroud, P., Ullman, S. and *et al.* (2014), 'Anatomy of hierarchy: Feedforward and feedback pathways in macaque visual cortex', *Journal of Comparative Neurology* **522**(1), 225–259.

May, R. M. (1976), 'Simple mathematical models with very complicated dynamics', *Nature* **261**(5560), 459–467.

Mehta, P., Bukov, M., Wang, C. H., Day, A. G., Richardson, C., Fisher, C. K. and Schwab, D. J. (2019), 'A high-bias, low-variance introduction to machine learning for physicists', *Physics Reports* **810**, 1–124.

Menczer, F., Fortunato, S. and Davis, C. A. (2020), *A First Course in Network Science*, Cambridge University Press.

Motter, A. E., Myers, S. A., Anghel, M. and Nishikawa, T. (2013), 'Spontaneous synchrony in power-grid networks', *Nature Physics* **9**(3), 191–197.

Newman, M. E. J. (2010), *Networks: an introduction*, Oxford University Press, Oxford; New York.

Nitzan, M., Casadiego, J. and Timme, M. (2017), 'Revealing physical interaction networks from statistics of collective dynamics', *Science advances* **3**(2), e1600396.

Novaes, M., dos Santos, E. R. and Pereira, T. (2021), 'Recovering sparse networks: Basis adaptation and stability under extensions', *Physica D: Nonlinear Phenomena* **424**, 132895.

Pecora, L. M. and Carroll, T. L. (1998), 'Master stability functions for synchronized coupled systems', *Physical Review Letters* **80**, 2109–2112.

Pereira, T., Eroglu, D., Bagci, G. B., Tirnakli, U. and Jensen, H. J. (2013), 'Connectivity-driven coherence in complex networks', *Physical Review Letters* **110**(23), 234103.

Pereira, T., van Strien, S. and Tanzi, M. (2020), 'Heterogeneously coupled maps: Hub dynamics and emergence across connectivity layers', *Journal of the European Mathematical Society* **22**, 2183–2252.

Pikovsky, A., Rosenblum, M. and Kurths, J. (2001), *Synchronization - A Universal Concept in Nonlinear Sciences*, Vol. 12 of *Cambridge Nonlinear Science Series*, Cambridge University Press.

Preyer, A. J. and Butera, R. J. (2005), 'Neuronal oscillators in aplysia californica that demonstrate weak coupling in vitro', *Physical Review Letters* **95**, 138103.

Rakshit, S., Ray, A., Bera, B. and Ghosh, D. (2018), 'Synchronization and firing patterns of coupled rulkov neuronal map', *Nonlinear Dynamics* **94**(2), 785–805.

Ren, J., Wang, W. X., Li, B. and Lai, Y. C. (2010), 'Noise bridges dynamical correlation and topology in coupled oscillator networks', *Physical Review Letters* **104**, 058701.

Rulkov, N. F. (2002), 'Modeling of spiking-bursting neural behavior using two-dimensional map', *Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics* **65**, 9.

Schultz, P., Peron, T., Eroglu, D., Stemler, T., Ramírez Ávila, G. M., Rodrigues, F. A. and Kurths, J. (2016), 'Tweaking synchronization by connectivity modifications', *Physical Review E* **93**(6), 062211.

Shandilya, S. G. and Timme, M. (2011), 'Inferring network topology from complex dynamics', *New Journal of Physics* **13**, 013004.

Stankovski, T., Pereira, T., McClintock, P. V. and Stefanovska, A. (2017), 'Coupling functions: universal insights into dynamical interaction mechanisms', *Reviews of Modern Physics* **89**(4), 045001.

Strogatz, S. H. (2000), *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry and Engineering*, Westview Press.

Tibshirani, R. (1996), 'Regression shrinkage and selection via the lasso', *Source: Journal of the Royal Statistical Society. Series B (Methodological)* **58**, 267–288.

Timme, M. (2007), 'Revealing network connectivity from response dynamics', *Physical Review Letters* **98**, 224101.

Timme, M. and Casadiego, J. (2014), 'Revealing networks from dynamics: An introduction', *Journal of Physics A: Mathematical and Theoretical* **47**, 343001.

Vogelstein, J. T., Perlman, E., Falk, B., Baden, A., Roncal, W. G., Chandrashekhar, V., Collman, F., Seshamani, S., Patsolic, J. L. and *et al.*, K. L. (2018), 'A community-developed open-source computational ecosystem for big neuro data', *Nature Methods* **15**, 846–847.

Wang, Q., Chen, G. and Perc, M. (2011), 'Synchronous bursts on scale-free neuronal networks with attractive and repulsive coupling', *PLoS one* **6**(1), e15851.

Wang, W.-X., Lai, Y.-C. and Grebogi, C. (2016), 'Data based identification and prediction of nonlinear and complex dynamical systems', *Physics Reports* **644**, 1–76.

Wang, W.-X., Lai, Y. C., Grebogi, C. and Ye, J. (n.d.), 'Network reconstruction based on evolutionary-game data via compressive sensing'.

Wang, W. X., Yang, R., Lai, Y. C., Kovanis, V. and Grebogi, C. (2011), 'Predicting catastrophes in nonlinear dynamical systems by compressive sensing', *Physical Review Letters* **106**.

Wang, W. X., Yang, R., Lai, Y. C., Kovanis, V. and Harrison, M. A. F. (2011), 'Time-series-based prediction of complex oscillator networks via compressive sensing', *EPL* **94**, 48006.

Watanabe, S. and Strogatz, S. H. (1994), 'Constants of motion for superconducting josephson arrays', *Physica D: Nonlinear Phenomena* **74**(3-4), 197–253.

Werner, G. (2010), 'Fractals in the nervous system: Conceptual implications for theoretical neuroscience', *Frontiers in Physiology* **1**, 15.

Winfree, A. T. (2001), *The Geometry of Biological Time*, Vol. 12, Springer Science & Business Media.

Yu, D., Righero, M. and Kocarev, L. (2006), 'Estimating topology of networks', *Physical Review Letters* **97**, 188701.

CURRICULUM VITAE

Name Surname: İrem TOPAL KEMENT

Education:

- **B.Sc.:** 2014, Istanbul Technical University, Physics Engineering Department
- **M.Sc.:** 2017, Istanbul Technical University, Physics Engineering Department
- **Ph.D.:** 2019, Istanbul Technical University, Physics Engineering Department (Courses transferred)
- **Ph.D.:** 2023, Kadir Has University, Computational Sciences and Engineering Department

Professional Experience:

- 2018-2020 Data Scientist at Miletos Inc., Istanbul.
- 2020-2023 Research Assistant at Kadir Has University, Istanbul.

Publications and Presentations Derived from the Thesis:

- **Topal I.**, Eroglu D. (2023). Reconstructing Network Dynamics of Coupled Discrete Chaotic Units. *Physical Review Letters,* 130, 117401.
- **Topal, I.**, Eroglu D. (2022). Reconstructing network dynamics of weakly coupled chaotic units from data (Talk), *Inverse Network Dynamics, Max Planck Institute for the Physics of Complex Systems*, Dresden, Germany.
- **Topal I.** (2021). Introduction to Network Sciences (Online Lecture), *19th Physics Days,* Istanbul.
- **Topal I.**, Eroglu D. (2021). Network Reconstruction from Data (Poster), *The EPS conference on Statistical Physics of Complex Systems,* ICTP, Trieste.
- **Topal I.**, Eroglu D. (2021). Reconstructing Network Dynamics of Weakly Coupled Chaotic Units (Online), *SIAM Conference on Applications of Dynamical Systems.*

- **Topal, I.** (2021). Synchronization of Chaotic Systems (Online Lecture), *17th Physics Days,* Istanbul.
- **Topal I.**, Eroglu D. (2020).Sensing Network Dynamics of Weakly Coupled Chaotic Units (Online), *Dynamics Days Digital.*

Other Publications and Presentations

- Topal, I., Servantie, J. (2019). Molecular Dynamics Study of the Thermal Conductivity in Nanofluids. *Chemical Physics* 516, 147-151.
- Topal, I., Kabakçıoğlu A., Subaşı, A.L., (2018). Prediction of DNA Binding Sites for a Transcription Factor (Poster), *Machine Learning Summer School at Bosphorus (BYOYO)* Istanbul.
- Topal, I. (2018). Molecular Dynamics Study of the Thermal Conductivity in Nanofluids (Talk), *The annual Statistical Physics Days,* İzmir.